D. Classroom Equipment ......................................................................................................................... 3
E. Faculty Offices ........................................................................................................................................ 4
F. Computing Facilities ............................................................................................................................... 4
G. Student Access ....................................................................................................................................... 10
H. Faculty Access ......................................................................................................................................... 11

8. Support .................................................................................................................................................. 1
   A. Faculty Stability ...................................................................................................................................... 1
   B. Faculty Professional Activities ............................................................................................................ 2
   C. Administration Effectiveness ............................................................................................................... 3
   D. Adequacy of Resources ....................................................................................................................... 4
   E. Continuity of Institutional Support ....................................................................................................... 5

APPENDIX I – Institutional Summary ......................................................................................................... 1

The Institution ............................................................................................................................................ 1
Type of Control ......................................................................................................................................... 1
History of Institution ................................................................................................................................ 1
Student Body .............................................................................................................................................. 1
Admissions Process .................................................................................................................................. 2
Regional or Institutional Accreditation ......................................................................................................... 3
Personnel and Policies ............................................................................................................................... 3
Academic Organizational Structure ............................................................................................................. 6
University Academic Structure .................................................................................................................. 7
Credit Unit .................................................................................................................................................. 8
Instructional Modes .................................................................................................................................. 8
Grade-Point Average and Graduation Requirements .................................................................................... 8
Academic Supporting Units ....................................................................................................................... 8
Non-Academic Supporting Units ................................................................................................................ 8
Table 1-1. History of Admissions Standards for Freshmen Admissions for Past Five Years ....................... 23
Table 1-2. Transfer Students for Past Five Academic Years ..................................................................... 23
Table 1-3. Undergraduate Enrollment Trends for Past Five Academic Years ....................................... 23
Table 1-4. Program Graduates .................................................................................................................. 24
Table 1-5. Personnel and Students ............................................................................................................ 25
Table 1-6. Program Enrollment and Degree Data ...................................................................................... 26
Table 1-7. Faculty 9-Month Salary Data ..............................................................27
APPENDIX II – Computer Science Department Policies and Procedures ............1
   CS Teaching Guidelines ..................................................................................1
   CS Program of Study Checklist ......................................................................5
   Official Academic Audit (sample).................................................................8
   Undergraduate Advising Hints .......................................................................9
APPENDIX III – Assessment Materials...............................................................1
   IIT Computer Science Undergraduate Program Evaluation Process...............1
   CS Course Student Assessments ...................................................................2
   CS Faculty/Course Self-Assessment ..............................................................5
   Communication Across the Curriculum Assessment 2006 .............................6
APPENDIX IV – Facilities ..................................................................................1
0. Preface

This section presents a complete outline of the material to be provided in each Self-Study Report. Each report should be formatted similarly to this section, preferably with the same heading titles. DO NOT DUPLICATE THE DETAILED INSTRUCTIONS.

A. Background Information

Please provide the following background information.

1. Degree Title

Give title(s) of all degrees awarded for the program under review, including options, etc., as specified in transcripts and/or diplomas, and describe as necessary.

| Bachelor of Science in Computer Science |

2. Program Modes

Indicate the modes, e.g., day, co-op, off-campus, on line, distance education, in which this program is offered and describe any differences in the information given for the computing unit as a whole in the Appendix I.

| The Bachelor of Science in Computer Science is a full-time, on-campus program. Most of the senior level classes are available off-campus at remote live TV locations or via the Internet, but this is to support traditional college students who in their third and fourth year and work part-time. The program is not advertised as a remote or Internet degree. |

3. Actions to Correct Previous Deficiencies, Weaknesses and/or Concerns

If specific program deficiencies, weaknesses and/or concerns were identified by the CAC during the most recent evaluation (visit or report), please refer to them and indicate the actions taken. Deficiencies, weaknesses and/or concerns that were addressed in the previous evaluation as being common to all computing programs should be addressed in each Self-Study Report.

| No specific deficiencies or weaknesses were identified during our most recent evaluation. However, three concerns were noted. These are addressed below: |

1) There is a need to formalize the procedure reviewing and implementing suggestions for course and program improvement (Standard I-5).

Our procedure for reviewing and implementing suggestions for course and program improvement is now much more substantial than existed previously. Essentially, one course is reviewed in detail every semester and feedback from all courses is discussed by the Undergraduate Studies Committee. This feedback inspires changes to the curriculum. Section 4 of this document describes the procedure in detail.

2) Advising duties must be recognized as part of faculty members' workload (Standard III-9).
We have clearly worked to improve our advising process. The reality is that for graduate studies all tenure-track faculty play a significant advising role. For undergraduate faculty, Professor Matt Bauer is the primary advisor for these students and to ensure continuity and stability his work is well documented and the Undergraduate Studies Committee members are fully able to assist students in cases where Professor Bauer is not available. Students often praise Professor Bauer’s quick response to their questions as well as his accessibility. He fields countless e-mails and responds to them routinely within a few hours and almost always within a day. Professor Bauer is the initial person who contacts students when they are admitted to our program and he assists them throughout their time at IIT. Professor Bauer is given a course reduction to ensure that he has adequate time to fully advise our undergraduate students. More details of our advising process can be found in Section 1.

3) There are an insufficient number of reasonably sized faculty offices (Standard VII-5).

This certainly a Departmental concern, however, our faculty are able to perform their duties. All tenure track faculty now have their own offices, which was not the case 6 years ago.

4. Contact Information
Identify the primary pre-visit contact person, e.g., the program chair or his/her designee if applicable. Provide name, address, telephone number, and e-mail address. Explicitly note any differences with the information on your Request for Evaluation (RFE).

Dr. Bogdan Korel
Illinois Institute of Technology
Computer Science Department
10 W. 31st Street
Chicago, IL  60616
312-567-5145
korel@iit.edu

B. Accreditation Summary
This section is the focus of the Self-Study Report. A complete description of how the program satisfies all of the requirements for each criterion must be presented. It is suggested that the information presented for each criterion be as complete as possible such that the Program Evaluator(s) can determine if all of the requirements are being met without cross-referencing material provided under other criteria. This may require some duplication of material but it should aid the Program Evaluator(s). Reference to the material provided in the Appendix I (found at the end of this document), and to other information provided by the institution, should be made as needed.

If you are having a program evaluated that exists on separate campuses, the answers to these questions may vary from one campus to another. If this is the case, please use
separate copies of this section for each campus, and clearly delineate which campus is being described.

The Computer Science Undergraduate Studies Committee, appointed by the Chair of the Computer Science Department, compiled this report. Comments and additional input were solicited from all faculty members. Faculty members assigned as Course Managers also worked with our committee in preparing this document. This was done in cooperation with all full-time and part-time faculty teaching the course during the 2007-2008 school year. Faculty resumes were updated during the Spring 2008 semester.

The CS program exists primarily on the main campus at IIT. Some courses are offered at the Rice campus and at other remote facilities. However, all tenure track faculty reside in the main campus facility and all required courses are taught in the main campus facility. Hence, all answers are focused on the main campus facility, but whenever it is relevant, external facilities are certainly included in the answers.

**Criterion 1. Students**

_Students can complete the program in a reasonable amount of time._

The IIT Computer Science Department ensures that all required Computer Science courses are offered every semester, and most elective Computer Science courses (both undergraduate and graduate) are offered every semester (a few are every other semester). All required non-Computer Science courses are offered every semester, and there is a wide selection of elective non-Computer Science courses offered every semester.

_Students have ample opportunity to interact with their instructors._

All faculty are required to have two to four office hours per week and faculty are requested to reply promptly to student e-mails. Additionally, most faculty take advantage of the BlackBoard system (see Section 7 for more details), which includes an electronic discussion board. All of our Computer Science courses also have teaching assistants assigned who also have office hours.

_Students are offered timely advising, by qualified individuals, about the program’s requirements and their career alternatives._

The Computer Science Department focuses its undergraduate advising efforts on Professor Matthew Bauer. He has been involved with the CS program for over fifteen years. He is the primary advisor for undergraduates and it is clear from numerous different feedback mechanisms that he is an outstanding advisor. Additionally, the department offers an Employment Seminar every semester with our ACM chapter in which speakers are brought in who are usually recent graduates with first hand knowledge of career alternatives.

_Students who graduate from the program meet all program requirements._

Educational Services at IIT implements a complete academic audit verifying all degree
Criterion 2. Program Educational Objectives

*The program has documented, measurable educational objectives that are based on the needs of the program’s constituencies.*

Our Program Objectives are constructed based on frequent discussions with faculty, students, government and industry sponsors, and alumni. The objectives are designed with our assessment strategy in mind. Hence, they are all measurable and we measure them at numerous opportunities. Standard measurement tools, which are detailed later in Section 2, include student course evaluations, faculty course evaluations, and alumni surveys.

The program objectives are:

A. Robust problem-solving skills.
B. Substantial knowledge of a broad class of problem-solving techniques (e.g., this includes Algorithms, heuristics, and design techniques).
C. Substantial understanding of the fundamentals of Computer Science.
D. Ability to clearly communicate technical concepts both orally and in writing
E. Ability to readily work with other disciplines.
F. Appropriate, occasional innovation of our curriculum so it incorporates ever-changing Computer Science technology.

Clearly robust problem solving skills and the ability to work with other disciplines is crucial to all of our stakeholders. Faculty members at IIT feel strongly that we must focus on the fundamentals of Computer Science and not be led simply by industry trend on particular products. Communication skills are a central skill that is clear in the University mission statement. Our Department goes beyond basic University requirements in this area as we insist students take Technical Writing (COM421) or Public Speaking (COM428) and we encourage them to take both. We note that our Departmental goal is one that maintains a constantly undated curriculum. There is clear proof of that simply by listing the key changes that have occurred since our last ABET evaluation:

- NSF sponsored development of undergraduate Data Mining and Information Retrieval courses. These courses have been transferred to leading universities worldwide and are taught there regularly.
- Two new specializations in “Information Security” and “Information and Knowledge Management Systems.”
- Since our last ABET visit, the department separated non-majors from majors in our Introductory Programming sequence to incorporate object-oriented theory earlier for our majors.
- Our Computer Organization and Systems Programming courses have been completely redesigned to update them to include modern technology.
- Finally, we have completed a “mock course” via an Interprofessional Project
(IPRO) to test new introductory programming paradigms. A new textbook has recently been developed by IIT Faculty to support that effort.

**Criterion 3. Program Outcomes**

*The program has documented, measurable outcomes that are based on the needs of the program’s constituencies.*

Our Program Outcomes were taken directly from the list provided in the Criterion for Section 3 of this document. We were able to clearly map these outcomes to course objectives found with the courses in our program. We have found that, in many cases, we achieve a single outcome with numerous course objectives. This gives us confidence that students will graduate with a firm set of tools embodied in these outcomes.

Our departmental outcomes are:

a. An ability to apply knowledge of computing and mathematics appropriate to the discipline
b. An ability to analyze a problem, and identify and define the computing requirements appropriate to its solution
c. An ability to design, implement and evaluate a computer-based system, process, component, or program to meet desired needs
d. An ability to function effectively on teams to accomplish a common goal
e. An understanding of professional, ethical, legal, security, and social issues and responsibilities
f. An ability to communicate effectively with a range of audiences
g. An ability to analyze the local and global impact of computing on individuals, organizations and society
h. Recognition of the need for, and an ability to engage in, continuing professional development
i. An ability to use current techniques, skills, and tools necessary for computing practices.
j. An ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices
k. An ability to apply design and development principles in the construction of software systems of varying complexity.
l. Be prepared to enter a top-ranked graduate program in Computer Science.

We include an additional outcome that indicates that we want our students to be eligible to enter a top-ranked CS program is one that our faculty certainly feels strongly about. We have seen our graduates obtain admission to top programs such as Georgia Tech, Stanford, Columbia, and Carnegie Mellon, and with the continued updates to our curriculum and our continuous improvement process, we expect to see even more students enter top programs in the future.
Criterion 4. Continuous Improvement

The program uses a documented process incorporating relevant data to regularly assess its program educational objectives and program outcomes, and to evaluate the extent to which they are being met.

The essence of our process is that the Undergraduate Studies Committee collects and reviews results of:

- Student course assessments
- Faculty Assessments
- Course Manager Assessments
- Alumni Surveys
- ACM Student Feedback seminars

To obtain indication of potential areas for improvement, every semester the committee chooses a course to review in depth. After reviewing course materials, the committee makes recommendations for improvement. Since our last ABET evaluation, the following courses have been reviewed in detail.

- CS330 Discrete Mathematics (Fall 2003)
- CS350 Computer Organization (Spring 2004)
- CS430 Introduction to Algorithms (Fall 2004)
- CS201 Accelerated Introduction to Programming, also CS115/CS116 (Spring 2005)
- CS450 Operating Systems (Fall 2005)
- CS487 Software Engineering (Spring 2006)
- CS331 Data Structures (Fall 2006)
- CS100 Introduction to the Profession (Spring 2007)

In every case, meaningful changes were suggested and implemented by our committee. Our key ongoing process can be summarized as:

1. Use raw data to identify a key focus area
2. Conduct a thorough review of the focus area (lecture notes, syllabus, assignments, completed coursework, etc)
3. Identify potential improvements and discuss them with the Course Manager
4. Implement improvements (if necessary)

Criterion 5. Curriculum

The program’s requirements are consistent with its educational objectives and are designed in such a way that each of the program outcomes can be achieved.

In Section 3, we show how the outcomes support the objectives for the program, and that each objective is supported by more than one outcome. In Section 5, we show a mapping between outcomes and the courses that support them. In many cases, more than one course meets an outcome. Hence, our program requirements go beyond simply satisfying
educational objectives and reinforce many of them through repetition found in the core program.

*The curriculum combines technical and professional requirements with general education requirements and electives to prepare students for a professional career and further study in the computing discipline associated with the program, and for functioning in modern society.*

Our curriculum satisfies the technical requirements. Our CS curriculum requires 85 credits of Computer Science, Math and Science course while the stated minimum is 70 credits. Our general education requirements include 27 credits of Humanities, Psychology, Interprofessional Projects, etc.

*The technical and professional requirements include at least one year of up-to-date coverage of fundamental and advanced topics in the computing discipline associated with the program.*

Our technical and professional requirements include 30 credits of advanced CS courses. This goes well beyond a single year of coverage and is nearly double the 16 credit stated minimum.

*In addition, the program includes mathematics appropriate to the discipline beyond the precalculus level.*

Our program requires three semesters of calculus, and three additional Math courses are required. The first is a choice between MATH332 and MATH333 (Matrices or Matrices with Complex Variables). The second is a choice between MATH474 and MATH475 (Probability and Statistics or Probability). Finally, a third math elective is required which is a choice between several CS relevant Junior and Senior level math courses.

*For each course in the major required of all students, its content, expected performance criteria, and place in the overall program of study are published.*

Every required course, its content, performance criteria, and place in the program are all published on the departmental Web site at
http://www.iit.edu/csl/cs/programs/undergrad/
http://www.iit.edu/csl/cs/programs/course_descriptions.shtml

**Criterion 6. Faculty**

* A. Faculty Qualifications

*Faculty members teaching in the program are current and active in the associated computing discipline. They each have the educational backgrounds or expertise consistent with their expected contributions to the program.*

*Each has a level of competence that normally would be obtained through graduate work*
in the discipline, relevant experience, or relevant scholarship.

All full-time tenure track faculty hold a Ph.D. in Computer Science and Engineering and many of them are frequently publishing papers in key publications, pursuing advanced research agenda and participating in as well as organizing significant conferences in their field. A review of Section 6 includes a long list of outstanding accomplishments among our faculty. Most non-tenure track full-time faculty members also have a Ph.D. and those that do not hold a M.S. in Computer Science. Our full time instructors are often recognized for excellence in teaching at the departmental and university level. The following University teaching awards have recently been bestowed upon our faculty:

- College of Science and Letters, Dean’s Excellence award in Teaching – Dr. Nazli Goharian (2005)

Collectively, they have the technical breadth and depth necessary to support the program.

Our faculty include specialists in every area of Computer Science including Algorithms (Reingold is a well recognized researcher in the field who has published a number of respected books on the topic, Kapoor, and Calinescu). Computer Networking includes Chlebus, Hood, Li, Roberson and Wan who has a tremendous number of significant journal publications. Frieder, an endowed chair who is a fellow of the ACM, IEEE, and AAAS leads an Information Retrieval Lab, which includes Argamon (A.I.), Goharian, Grossman, and Yee. Operating Systems and Distributed Systems has a group led by (Sun) and includes Leung, Ren and Lan. Agam concentrates on Computer Graphics and Vision, and Korel’s area is Software Engineering. More importantly, the amount of collaboration between these groups is significant with clear examples being found in papers on Distributed Peer-to-Peer Information Retrieval, etc.

B. Faculty Size and Workload

There are enough full-time faculty members to provide continuity, oversight, and stability, to cover the curriculum reasonably, and to allow an appropriate mix of teaching, professional development, scholarly activities, and service for each faculty member.

The 16 tenure track faculty members have expertise that covers the core areas of the curriculum. Teaching loads are divided between tenure-track and non-tenure-track faculty, with tenure-track faculty overseeing course development and delivery. There is a reasonable balance here as the tenure track faculty members often teach junior and senior level courses and Interprofessional Projects, where they are able to insert recent advancements in their fields not already in the curriculum. All faculty members also participate in research or service activities.

The faculty assigned to the program has appropriate authority for the creation, delivery,
evaluation, and modification of the program, and the responsibility for the consistency and quality of its courses.

Every undergraduate course has a faculty member assigned as Course Manager, who is responsible for ensuring the consistency and quality of the courses.

**Criterion 7. Facilities**

*Institutional facilities including the library, other electronic information retrieval systems, computer networks, classrooms, and offices are adequate to support the educational objectives and outcomes of the program.*

The library is described in detail in Section 7. The library provides access to numerous electronic journals and proceedings include the ACM Digital Library and the IEEE digital library. Classrooms in the Computer Science Department have recently been upgraded to provide streamline use of video and image content and are certainly sufficient to meet our objectives.

*Computing resources are available, accessible, systematically maintained and upgraded, and otherwise adequately supported to enable students to achieve the program’s outcomes and to support faculty teaching needs and scholarly activities.*

Both the Office of Technology Services (OTS) and the Computer Science Department provide on-campus lab facilities, which are used to support laboratory classes in which students work hands-on with a machine while a laboratory instructor leads them through various drills. The labs are periodically upgraded.

*Students and faculty members receive appropriate guidance regarding the computing resources and laboratories available to the program.*

Computing labs are discussed in each relevant course. Additionally, if students work with one of our research groups, they obtain access to even more sophisticated compute power. For example, our Information Retrieval Lab was just given a $300K 28 processor, 3 TB, high-performance database machine from a startup named Netezza.

**Criterion 8. Support**

*The institution’s support for the program and the financial resources available to the program are sufficient to attract and retain qualified faculty members, administer the program effectively, acquire and maintain computing resources and laboratories, and otherwise provide an environment in which the program can achieve its educational objectives and outcomes.*

Retention among instructors and faculty is extremely high. The only tenure-track faculty who have not stayed with our Department since our last accreditation are those who were not granted tenure or retired. Our Department has been able to maintain small class sizes and an active research program.
Support and resources are sufficient to provide assurance that the program will retain its strength throughout the period of accreditation.

Given the track record of our faculty, we expect the quality of both research and teaching to continue to improve.

Criterion 9. Program Criteria
Each program must satisfy applicable Program Criteria (if any). Program Criteria provide the specificity needed for interpretation of the General Criteria as applicable to a given discipline. If a program, by virtue of its title, becomes subject to two or more sets of Program Criteria, then that program must satisfy each set of Program Criteria; however, overlapping requirements need to be satisfied only once.

3. Program Outcomes
The program enables students to achieve, by the time of graduation:

(j) An ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices. [CS]
(k) An ability to apply design and development principles in the construction of software systems of varying complexity. [CS]

We have included these two additional outcomes. Section 3 and Section 5 describe how our courses support these outcomes as evaluated by our measurement tools.

5. Curriculum
Students have the following amounts of course work or equivalent educational experience:

a. Computer science: One and one-third years that includes:

1. coverage of the fundamentals of algorithms, data structures, software design, concepts of programming languages and computer organization and architecture. [CS]

    We require specific courses on algorithms (CS430), data structures (CS331), software engineering (CS487), Programming Languages (CS440), Computer Organization and Assembly Language (CS350), and architecture is covered in the introductory courses and operating systems (CS450).

2. an exposure to a variety of programming languages and systems. [CS]

    Students who complete our program are certainly exposed to Java in our introductory courses, Assembler in CS350, and C/C++ in CS350/CS351. Our Programming Languages course (CS440) is designed to expose students to a variety of programming languages.
3. proficiency in at least one higher-level language. [CS]

Students are required to complete advanced programming projects in many 4xx level courses where the language of choice is either C++, Java or Ruby.

4. advanced course work that builds on the fundamental course work to provide depth. [CS]

Depth is provided with required CS4xx classes in Algorithms, Programming Languages, Operating Systems and Software Engineering.

b. One year of science and mathematics:

1. Mathematics: At least one half year that must include discrete mathematics. The additional mathematics might consist of courses in areas such as calculus, linear algebra, numerical methods, probability, statistics, number theory, geometry, or symbolic logic. [CS]

In addition to the three required courses in Calculus, we require Discrete Math (CS330 or MATH230), Linear Algebra (MATH332 or MATH333) and Probability/Statistics (MATH474 or MATH475). One additional elective is chosen from upper level Math classes such as Graph Theory, Combinatorics, Number Theory, and Differential Equations.

2. Science: A science component that develops an understanding of the scientific method and provides students with an opportunity to experience this mode of inquiry in courses for science or engineering majors that provide some exposure to laboratory work. [CS]

In additional to two, four-credit Physics courses being required (with Labs), two other science electives are also required chosen from Chemistry, Biology, Materials Science or Psychology (scientific methods).

6. Faculty Qualifications

Some full time faculty members have a Ph.D. in computer science.

All full time tenure track faculty hold a Ph.D. in Computer Science and numerous full-time instructors also hold a Ph.D. in Computer Science.
1. Students

Criterion
Students can complete the program in a reasonable amount of time. They have ample opportunity to interact with their instructors. Students are offered timely advising, by qualified individuals, about the program’s requirements and their career alternatives. Students who graduate from the program meet all program requirements.

A. Frequency of Course Offerings

1. List below the course numbers, titles, semester hours and frequency of offerings for all courses required for the major that are offered less frequently than once per year.

<table>
<thead>
<tr>
<th>Course</th>
<th>Title</th>
<th>Semester Hours</th>
<th>Frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>None. All of the required courses (both CS and non-CS) for the BS in CS are offered at least once per year.</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

2. Explain how it is determined when each required course will be offered, e.g., rotation, odd-numbered years, etc

All required courses (both CS and non-CS) are offered every semester. Only CS100 is offered once per year in the Fall, because it is designed exclusively for new Freshman Computer Science students.

3. List below the course numbers, titles, and semester hours of courses allowed for the major but not required (i.e., electives within the major), and explain how it is determined when they will be offered.

**Computer Science Electives** (five required):

- Five courses at (4xx or 5xx) level Computer Science Electives (15 hours) are required for the CS major. At least one must be from CS422, CS425, CS429, CS447, and CS470. The remaining can be any CS4xx or CS5xx course, or one of ECE218 Digital Systems, ECE441 Microcomputers, ECE449 Object-Oriented Programming and Computer Simulation.
- The following four electives are offered every semester:
  - CS425-Database Organization
  - CS441-Current Topics in Programming Languages
  - CS445-Object-Oriented Design and Programming
  - CS455-Data Communications
- The following 4xx level CS elective courses are offered at least every other semester:
  - CS422-Data Mining
  - CS429- Information Retrieval
  - CS447-Distributed Objects
  - CS458- Information Security
  - CS470-Computer Architecture
• The following 4xx level CS elective courses are offered at least every third semester:
  ▪ CS411-Computer Graphics
  ▪ CS480-AI:Planning and Control
  ▪ CS481-AI:Language Understanding
  ▪ CS482-Information and Knowledge Management Systems
• 15-20 5xx level CS elective courses are offered every semester: Offerings are determined by the Computer Science Department based on historical demand and current enrollment trends as well as faculty preferences.

Math Electives (one required):

• The following are offered every semester.
  ▪ MATH252 Differential Equations
• The following are offered every other semester
  ▪ MATH410-Number Theory
  ▪ MATH435-Linear Optimization
  ▪ MATH453-Combinatorics
  ▪ MATH454-Graph Theory
  ▪ MATH476-Probability
  ▪ MATH482-Intro to Markov Processes
• Offerings are determined by the Mathematics Department based on historical demand and current enrollment trends as well as faculty preferences.

Science Electives (two required):

• The following are offered every semester.
  ▪ CHEM122/124-Principles of Chemistry
  ▪ MS201-Material Science
  ▪ PHYS223-General Physics III: Thermal and Modern Physics
• The following are offered every other semester
  ▪ BIOL107-General Biology Lectures
  ▪ BIOL115-Human Biology Chemistry
• The following are offered every third semester
  ▪ PHYS120-Astronomy
  ▪ PSYC204-Experimental Psychology and Research Methods
• Offerings are determined by the Science Department based on historical demand and current enrollment trends as well as faculty preferences.

Humanities Electives

• One at 1xx level Humanities Elective (3 hours)
  ▪ All three of these “no prerequisite” courses are offered every semester (HUM102-Industrial Culture, HUM104-Age of Darwin, HUM106-Life Stories).
  ▪ At least one AAH1xx (Art History) and COM1xx (Language and
Culture) courses are offered every semester. These offerings are determined by the Humanities Department based on historical demand and current enrollment trends as well as faculty preferences.

- Three at 3xx level Humanities Electives (9 hours) – At least 10-12 courses that students have pre-requisites for are offered every semester (LIT-Literature, HIST-History, PHIL-Philosophy, COM-Communications)
- Offerings are determined by the Humanities Department based on historical demand and current enrollment trends as well as faculty preferences.

Social Science Electives

- Two at (1xx or 2xx) level Social Science Electives (6 hours) – At least one of these “no prerequisite” courses is offered every semester in each of these areas. Offerings are determined by the Social Science, Psychology and Business Departments based on historical demand and current enrollment trends as well as faculty preferences.
  - Political Science - PS200-American Government or PS201-Politics and Public Policy, and assorted other PS2xx courses.
  - Psychology - PSYC221-Human Behavior, Growth and Learning or PSYC222-Brain, Mind and Behavior
  - Sociology - SOC200-Introduction to Sociology, SOC201-Social Psychology, SOC203-Engaging Sociology, and assorted other SOC2xx courses.
  - Economics – ECON151-Macroeconomics, ECON152-Macroeconomics, ECON211-Economics
- Two at (3xx or 4xx) level Social Science Electives (6 hours) – At least 10-12 courses that students have pre-requisites for are offered every semester in these areas: PS-Political Science or PSYC-Psychology or SOC-Sociology or ECON-Economics. Offerings are determined by the Social Science, Psychology and Business Departments based on historical demand and current enrollment trends as well as faculty preferences.
- Offerings are determined by the Social Science, Psychology and Business Departments based on historical demand and current enrollment trends as well as faculty preferences.

Interprofessional Projects (two required):

- 40-50 different Interprofessional Projects (IPRO) (1-3 CS specific led by CS faculty) are offered every semester across a wide variety of topics and application areas. All IPROs are reviewed and approved by the IPRO Office. See http://ipro.iit.edu/home/main.php

Free Electives (four required):

- Students may choose any course for free electives except for remedial courses.
B. Interaction with Faculty

1. Describe how you achieve effective interaction between students and faculty.

Historically, the sizes of our 1xx, 2xx and 3xx level Computer Science courses is as follows:

- CS100 – 20-25 students, CS115 – 30-40 students, CS116 – 30-40 students
- CS201 – 30-40 students
- As to the 4xx level Computer Science courses the sizes of the classes are usually between 20 and 60 students. Most classrooms are small (20-35 students). Remote graduate students often register for 4xx classes as they begin their M.S. or Ph.D. studies.

As a rule, undergraduate students are not allowed to take distance learning IITV classes or IIT Online Internet classes. However, students that are in the live section of a class also offered over the Internet may review lectures on the Internet. All of our 4xx level courses have this option. Any of our courses that are offered in both day and night sections, the day sections are restricted for undergraduates only to ensure the undergraduate population can take the class with their peers.

In all levels of Computer Science courses, all faculty are required to have a reasonable number of office hours depending upon the size of the class, usually 2-4 office hours per week, and also must promptly reply to e-mail messages and questions posted on the discussion board forums on the blackboard. This is outlined in the “CS Teaching Guidelines” (See Appendix II Computer Science Department Policies and Procedures).

Our undergraduate student ACM Programming Competition teams are led by Professor Calinescu, and have performed very well of late, sending a team to the international competition twice in a row.

Finally, there are also social interactions available for students and faculty such as informal weekly lunches, end of semester parties, and the University’s Sophomore Leadership Retreat.

C. Student Advising

1. Describe your system of advisement for students on how to complete the program. Indicate how you ensure that such advisement is available to all students.

Advising of New Students
The Department has assigned all newly admitted CS undergraduate students (freshmen and transfers) to Prof. Matthew Bauer as their advisor. Prof. Bauer is contacted in May of each year with the list of newly admitted students. Shortly thereafter, he contacts the new
each year with the list of newly admitted students. Shortly thereafter, he contacts the new students, and advises them via e-mail, phone or in person before they arrive on campus. Usually 30 minutes to an hour is spent on each incoming student plus numerous e-mails. After this advising session the new student registers online for the Fall semester.

This workload is reasonable in that we have averaged fifty incoming students and Prof. Bauer is given a course reduction for his efforts and is compensated for this advising. This fills a key gap in that Faculty are on nine-month contracts and may have other obligations, such as teaching and research, during the summer.

Once students arrive on campus, advising continues. Prof. Bauer answers e-mail with no more than a 24-hour delay and often he e-mails back within four hours, and also meets personally with many students. Students frequently praise his efforts as an advisor. Prof. Bauer has documented our advising process in the “Undergraduate Advising Hints” (see Appendix II Computer Science Department Policies and Procedures). This document also has served as a model for other Departments on campus.

Advising of Continuing Students
Continuing students follow the CS Curriculum that was in place when he/she began at IIT. During a student’s (new student or transfer student) initial advising session with their advisor, they are informed of the current CS Curriculum, usually this is the one printed in the current IIT UG Bulletin. They are also given, and shown how to use, the current “CS Program of Study Checklist” (see Appendix II Computer Science Department Policies and Procedures). This is a form used internally by the Computer Science Department and CS advisors and updated, with the student, in each advising session. When a student has completed one year of study (or around 30 credit hours), the Office of Educational Services completes an “Official Academic Audit” (see Appendix II Computer Science Department Policies and Procedures) for the student’s chosen major and sends a copy to the student and Department. The “Official Academic Audit” is used by the Computer Science Department and CS advisors and updated, with the student, in each advising session. This audit is officially updated by Educational Services whenever the student requests it and also when the student is within a year of graduation. Finally, all students are informed of the “Undergraduate Advising Hints” (see Appendix II Computer Science Department Policies and Procedures) that explains the details of the CS Curriculum and the procedures for advising and registration. Any modifications to the CS Curriculum are immediately posted to the Computer Science Department Web page with appropriate effective dates.

Other Advising

- The past few years the Computer Science Department has sponsored a peer advising session every semester. Currently the Student Government Association is sponsoring peer advising for all Departments in conjunction with professional organizations for each major.
- Advisors are aware of the various student assistance services available on campus (Academic Resource Center, Student Health Center, and Counseling Center) and
advisors can also serve as liaisons between students and other faculty.

- As an additional preventative measure against students getting into trouble academically without their advisor being aware, Midterm Grades are submitted by instructors for all first year undergraduates in 1xx, 2xx and 3xx level courses. Freshmen Advisors receive Midterm Grades and all students receiving a C or worse grade are required to have advising appointments with their advisor to discuss the student’s performance.

- In the students’ first year, there is a school wide retention program. All attendance, assignment, homework, quiz grades are recorded every week and submitted to Prof. Bauer and other “learning assistants.” The “learning assistants” are junior and senior level undergraduate mentors that are assigned to students. The ARC (Academic Resource Center) is also copied on these results. Hence, Prof. Bauer checks for any problems and contacts students if he sees a potential pattern that could affect the student’s retention.

- Probation/Dismissal Advising – After every semester's grades are reported, the lead CS Undergraduate Advisor meets with the Associate Provost to discuss students being put on IIT Academic Probation/Dismissal (as defined in the Undergraduate Bulletin) and students that are close to being put on IIT Academic Probation. Students are required to have additional meetings with their Advisor, and/or the lead CS Undergraduate Advisor, and/or the Associate Provost. Sometimes credit hour limits on the next semester are imposed, or recommendations are made to take courses again, or an academic contract with the student is required. Whatever the case, the action is documented in the Computer Science Department student files and on the Student Information System (SIS, Banner in Fall 2008).

- The Department is considering a more formal mentor program of assigning students to full-time doctoral instructional and tenured/tenure-track faculty. For the last two years, the Computer Science Department has informal weekly lunches for faculty, staff and students. Frequently, this results in open discussion and mentoring by tenure track faculty.

2. When students need to make career choices, what is their procedure for obtaining advising? How do they have adequate access to qualified professionals when necessary?

The Career Management Center provides students with information on options that may match to the interest and area of the study of each student. The Career Management Center (CMC) is the liaison between our highly diverse talent pool of students and alumni and employers who benefit from their expertise. They are open Monday – Friday 9:00 AM - 5:00 PM.

ACM Student Chapter “ACM Employment Seminar” program every semester is another forum where students are able to meet with professionals and discuss career options.

Prof. Bauer also is available for career discussions and so students are aware of all the
resources available to them on campus. Furthermore, all faculty members have direct contact with the students and are available to answer student’s questions for making career choices.

3. Advising must be done by qualified individuals. Discuss the system by which advisors become qualified.

Clearly, Prof. Bauer is the key focal point for undergraduate advising for our Department. He brings a wealth of experience, 15 years, to the task. Because of the documentation and Prof. Bauer’s membership on the CS Undergraduate Studies Committee, all committee members are able to serve as advisors to undergraduates as a backup to Prof. Bauer.

D. Meeting the Requirements
1. Describe your standards and procedures for ensuring that graduates meet all of the requirements of the program.

The Office of Educational Services is responsible for certifying that an individual student has satisfied the prescribed curriculum for a Bachelor of Science degree. When necessary, the associate chair of the individual Department provides assistance in the verification process.

An academic audit provides a summary of a student’s academic status to date and lists the courses to be completed in order to receive a degree. Students who have completed at least 60 semester hours (including applicable transfer credit) will receive an audit from the Office of Educational Services. After receiving their first audit, students may request periodic updates. Faculty advisors have access to the same database of student information that is used by the Office of Educational Services.

After a student submits an application for graduation, a graduation audit is completed and a letter, which indicates the remaining requirements for the degree, is sent to the student. The final audit is completed when the grades for the semester are recorded and, if all requirements are completed, the degree is awarded.
2. Program Educational Objectives

Criterion
The program has documented, measurable program educational objectives that are based on the needs of the program’s constituencies.

1. Provide the institution’s mission statement. Include any other mission statements that are relevant.

IIT Mission
To advance knowledge through research and scholarship, to cultivate invention improving the human condition, and to educate students from throughout the world for a life of professional achievement, service to society, and individual fulfillment.

College of Science and Letters Mission
- Deliver superior educational and research opportunities through B.S., M.S. and Ph.D. degree programs as well as certificate, professional masters, and short-course programs.
- Provide responsive, appropriate core curriculum courses for students from all academic units at IIT.
- Engage in nationally and internationally recognized research and scholarship in biology, chemistry, computer science, mathematics and science education, humanities, mathematics, physics, and social sciences.
- Promote interdisciplinary and collaborative research among faculty and students within and outside of IIT and the college.

Department of Computer Science Mission
The IIT Department of Computer Science will be an exciting place to perform high-impact research and to learn about the latest developments in the constantly developing field of Computer Science. The department will ensure that students graduate knowing the fundamentals of Computer Science. We will be an excellent department as measured by:

- Quality, breadth, and impact of our research.
- Quality of teaching.
- Excellent preparation of our graduates for leadership in the profession and higher education.
- Dedication of our department staff to outstanding service, and our commitment to recognize everyone's contribution to our success.

2. List the program’s educational objectives. Explain how and where they are documented outside of this Self-Study.
Bachelor of Science in Computer Science Program Objectives

The Bachelor of Science in Computer Science program has been offered at IIT since 1971. Our goal has been and continues to be a high quality degree program that ensures that students will be able to integrate theory and practice, recognize the importance of abstraction and appreciate the value of efficient design created to meet clearly developed requirements. The program is intended to prepare students for lifelong learning as they undertake professional careers in computing.

Students will be able to solve problems using algorithms and techniques. They will have sufficient understanding of the theoretical underpinnings of Computer Science such that learning a new programming language, operating system, or information system will be viewed as a routine matter – something that can be done in 2-3 days. Additionally, students will graduate with the ability to communicate well, both orally and in writing. Students will graduate with the ability to work well in a multi-disciplinary environment. Finally, students will graduate with an understanding of the context of their skills within a broader academic and applied environment.

Specifically, the core objectives are to ensure that students graduate with:

A. Robust problem-solving skills.
B. Substantial knowledge of a broad class of problem-solving techniques (e.g.; this includes Algorithms, heuristics, and design techniques).
C. Substantial understanding of the fundamentals of Computer Science.
D. Ability to clearly communicate technical concepts both orally and in writing.
E. Ability to readily work with other disciplines.
F. Appropriate, occasional innovation of our curriculum so it incorporates ever-changing Computer Science technology.

All of these program objectives are published on the Computer Science departmental website http://www.iit.edu/csl/cs/about/mission.shtml

3. Describe how your program's educational objectives align with your institution's mission.

The program objectives align with the institution mission because the core mission of the university is to:

“To advance knowledge through research and scholarship, to cultivate invention improving the human condition, and to educate students from throughout the world for a life of professional achievement, service to society, and individual fulfillment.”

The Computer Science program objectives ensure that the curriculum stays current by insisting on “innovation of our curriculum.” Additionally, the university mission to “advance knowledge” is clearly met by the Computer Science objective to ensure that students learn the “theoretical underpinnings of Computer Science such that learning a
new programming language, operating system or information system will be viewed as a routine matter – something that can be done in two to three days.”

4. Explain how the program's educational objectives align with the needs of its constituencies, and include a list of the stakeholders. Also describe the role the constituencies played in formulating the educational objectives.

The key stakeholders of the IIT Computer Science program are the:

- Students
- Faculty
- Alumni
- Government and Industry Partners

The educational objectives align with each of these stakeholders.

**Students** clearly have an interest in learning the key underpinnings of Computer Science as this will prepare them for careers in Computer Science in industry, research careers in academia, or the problem solving skills will help them if they ultimately work with other disciplines. Students provide feedback on individual course evaluations.

**Faculty** are well served by ensuring that the foundations of Computer Science are taught as the students they work with will be capable of doing some undergraduate research and ultimately should be extremely valuable if they continue on in graduate work. Additionally, teaching such fundamentals is far more rewarding than simply teaching surface-level knowledge. The Undergraduate Studies committee works with course managers to review courses to ensure that are aligned with the departmental objectives.

**Alumni** are clearly influenced by the reputation of our department. This will help them advance their careers. They also may call upon faculty and students to help them with problems they face as their careers advance. Periodically, the department surveys alumni to confirm our objectives are in line with alumni views.

**Government and Industry Partners.** These partners rely upon students at IIT for new additions to their workforce. Clearly, the ability of these students to adapt to new technology is paramount. The department has several research labs that are supported by government and industry partners. Through numerous status briefings and final reports as well as feedback from these reports, these partners are able to provide input into our departmental objectives.
5. For each program educational objective, indicate the mechanism(s) used to measure it.

We first describe the core measurement tools used by the department (for measuring both objectives and outcomes).

1. **Student Course Assessment** (end of every semester) - Student surveys are given every semester by the undergraduate studies committee. They are geared specifically toward asking students how well the department is meeting objectives for each course. It is our belief that if we meet objectives for each course, that the overall departmental objectives will be met.

2. **Student IIT evaluations** - Students are asked to fill out an evaluation by the university at the end of each semester that essentially provides direct feedback on the quality of the professor who taught the course. Clearly, the undergraduate studies committee reviews these results to ensure that the quality of teaching is at a high level.

3. **Instructor Course Assessment** (end of every semester) - Faculty are surveyed at the completion of each course that is delivered to obtain their views on whether or not the course they taught met the core objectives of the course. It is also their opportunity to suggest changes for future renditions of the course.

4. **CS Undergraduate Committee Detailed Course Review** (one course every semester) - The CS Undergraduate Committee selects one required CS course each semester for a detailed review including all assignments, lecture materials, projects, and exams. This selection is based on recent student/faculty surveys.

5. **Recent Alumni Survey** - Alumni are surveyed each year by our department. In particular we send surveys to graduating students. We also talk to Alumni every semester at our ACM Employment Seminars to obtain their thoughts on the program.

6. **ACM Student Feedback Forum** - Periodically (usually once per year), the CS department sponsors an ACM Student Feedback Forum. Students are encouraged to give direct feedback on any portion of the CS program.

7. **IPRO Evaluation** - IPRO office asks students to evaluate IPRO objectives at the end of each semester. We plan to use this data to evaluate the effectiveness of a student teamwork, project management and communication skills.

8. **Communication Requirements** - The Communication Across the Curriculum Office does periodic review of “C” courses in each major. We use this data to evaluate the effectiveness of a student’s communication skills.

We will incorporate an additional measurement tool starting Fall 2008. At present, we rely on less formal, anecdotal conversations with employers and industry partners.

9. **Co-Op/Employer Survey** - Employers will be surveyed by IIT after each internship/Co-Op. The undergraduate studies committee will review these surveys to ensure that employers are generally pleased with IIT graduates.
The following table maps core objectives to evaluation methods that measure each objective. To simplify the table, the number corresponding to the measurement mechanism matches the numbers in the list given above. Note that only measurement tools 5, 6 and 7 are used for objectives measurement, as they are very broad evaluation mechanisms.

<table>
<thead>
<tr>
<th>Objective</th>
<th>Measurement Mechanism</th>
</tr>
</thead>
<tbody>
<tr>
<td>A. Robust problem-solving skills.</td>
<td>5,6,7</td>
</tr>
<tr>
<td>B. Substantial knowledge of a broad class of problem-solving techniques (e.g.; this includes Algorithms, heuristics, and design techniques).</td>
<td>5,6</td>
</tr>
<tr>
<td>C. Substantial understanding of the fundamentals of Computer Science.</td>
<td>5,6</td>
</tr>
<tr>
<td>D. Ability to clearly communicate technical concepts both orally and in writing</td>
<td>5,7</td>
</tr>
<tr>
<td>E. Ability to work with other disciplines</td>
<td>5,7</td>
</tr>
<tr>
<td>F. Appropriate, occasional innovation of our curriculum so it incorporates ever-changing Computer Science technology.</td>
<td>5,6</td>
</tr>
</tbody>
</table>
3. Program Outcomes

Criterion
The program has documented measurable outcomes that are based on the needs of the program's constituencies.

The program enables students to achieve, by the time of graduation:

(a) An ability to apply knowledge of computing and mathematics appropriate to the discipline;
(b) An ability to analyze a problem, and identify and define the computing requirements appropriate to its solution;
(c) An ability to design, implement and evaluate a computer-based system, process, component, or program to meet desired needs;
(d) An ability to function effectively on teams to accomplish a common goal;
(e) An understanding of professional, ethical, legal, security, and social issues and responsibilities;
(f) An ability to communicate effectively with a range of audiences;
(g) An ability to analyze the local and global impact of computing on individuals, organizations and society;
(h) Recognition of the need for, and an ability to engage in, continuing professional development;
(i) An ability to use current techniques, skills, and tools necessary for computing practices.

For computer science programs:

(j) An ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices;
(k) An ability to apply design and development principles in the construction of software systems of varying complexity.

For information systems programs:

The program outcomes are consistent with those accepted by the information systems community.

The program enables students to achieve the following attributes by the time of graduation:

An understanding of processes that support the delivery and management of information systems within a specific application environment.

For information technology programs:

(j) An ability to use and apply current technical concepts and practices in the core information technologies;
(k) An ability to identify and analyze user needs and take them into account in the selection, creation, evaluation and administration of computer-based systems;
(l) An ability to effectively integrate IT-based solutions into the user environment;
(m) An understanding of best practices and standards and their application;
(n) An ability to assist in the creation of an effective project plan.

1. List the program’s outcomes. Discuss how and where they are documented outside of this Self-Study.

**Computer Science Program Outcomes**

The department concurs with the all of the outcomes suggested in the ABET criteria.

a. An ability to apply knowledge of computing and mathematics appropriate to the discipline
b. An ability to analyze a problem, and identify and define the computing requirements appropriate to its solution
c. An ability to design, implement and evaluate a computer-based system, process, component, or program to meet desired needs
d. An ability to function effectively on teams to accomplish a common goal
e. An understanding of professional, ethical, legal, security, and social issues and responsibilities
f. An ability to communicate effectively with a range of audiences
g. An ability to analyze the local and global impact of computing on individuals, organizations and society
h. Recognition of the need for, and an ability to engage in, continuing professional development
i. An ability to use current techniques, skills, and tools necessary for computing practices.
j. An ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices
k. An ability to apply design and development principles in the construction of software systems of varying complexity.

In addition to the outcomes found in the ABET criteria, our Computer Science program has the following additional expected outcome:

l. Be prepared to enter a top-ranked graduate program in Computer Science.

These outcomes are documented externally on http://www.iit.edu/csl/cs/about/mission.shtml
2. For each program outcome, indicate the mechanism(s) used to measure it.

<table>
<thead>
<tr>
<th>Outcome</th>
<th>Measurement Mechanism</th>
</tr>
</thead>
<tbody>
<tr>
<td>a. An ability to apply knowledge of computing and mathematics appropriate to the discipline</td>
<td>1,2,3,4</td>
</tr>
<tr>
<td>b. An ability to analyze a problem, and identify and define the computing requirements appropriate to its solution</td>
<td>1,2,3,4</td>
</tr>
<tr>
<td>c. An ability to design, implement and evaluate a computer-based system, process, component, or program to meet desired needs</td>
<td>1,2,3,4</td>
</tr>
<tr>
<td>d. An ability to function effectively on teams to accomplish a common goal</td>
<td>3,7</td>
</tr>
<tr>
<td>e. An understanding of professional, ethical, legal, security, and social issues and responsibilities</td>
<td>3,7</td>
</tr>
<tr>
<td>f. An ability to communicate effectively with a range of audiences.</td>
<td>7,8</td>
</tr>
<tr>
<td>g. An ability to analyze the local and global impact of computing on individuals, organizations and society</td>
<td>3,7</td>
</tr>
<tr>
<td>h. Recognition of the need for, and an ability to engage in, continuing professional development</td>
<td>5</td>
</tr>
<tr>
<td>i. An ability to use current techniques, skills, and tools necessary for computing practices.</td>
<td>1,2,3,4</td>
</tr>
<tr>
<td>j. An ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices</td>
<td>1,2,3,4</td>
</tr>
<tr>
<td>k. An ability to apply design and development principles in the construction of software systems of varying complexity.</td>
<td>1,2,3,4</td>
</tr>
<tr>
<td>l. Be prepared to enter a top-ranked graduate program in Computer Science.</td>
<td>1,2,3,4,5</td>
</tr>
</tbody>
</table>

3. Explain the relationship between the outcomes and the needs of the program’s constituencies. Also explain the role played by the various constituencies in formulating the program outcomes.
The key stakeholders, whose role is to formulate the outcomes of the IIT Computer Science program, are:

- Students
- Faculty
- Alumni
- Government and Industry Partners

The educational outcomes clearly match the needs of the programs constituencies.

**Students** clearly have an interest in learning strong technical abilities as described in outcomes: a, b, c, i, j, and k. Students frequently desire to attend a quality graduate program and thus outcome l is of importance to them. These outcomes prepare students with the key underpinnings of Computer Science. The technical outcomes are not formulated based on student feedback.

**Faculty** are well served by ensuring that the foundations of Computer Science are taught as describe in outcomes a, b, c, i, j, and k. Additionally, teaching such fundamentals is far more rewarding than simply teaching surface-level knowledge. The Undergraduate Studies Committee works with course managers to review courses to ensure that are aligned with the departmental objectives.

**Alumni** are clearly interested in lifelong learning, and as such objective: g and h is of great interest to them. Periodically, the department surveys alumni to confirm our objectives are in line with alumni views and alumni are certainly able to indicate desired outcomes at this time.

**Government and Industry Partners.** These partners rely upon students at IIT for new additions to their workforce. Clearly, communication skills and teamwork (d and f) are of importance to these partners. The department has several research labs that are supported by government and industry partners. Through numerous status briefings and final reports as well as feedback from these reports, these partners are able to provide input into our departmental outcomes.
4. Indicate how your program outcomes map to your program educational objectives. The following table maps program objectives to outcomes.

<table>
<thead>
<tr>
<th>Objective</th>
<th>Outcome</th>
</tr>
</thead>
<tbody>
<tr>
<td>A. Robust problem-solving skills.</td>
<td>a,b,c,i,j,k</td>
</tr>
<tr>
<td>B. Substantial knowledge of a broad class of problem-solving techniques</td>
<td>a,b,c,i,j,k</td>
</tr>
<tr>
<td>(e.g.; this includes Algorithms, heuristics, and design techniques).</td>
<td></td>
</tr>
<tr>
<td>C. Substantial understanding of the fundamentals of Computer Science.</td>
<td>a,b,c,i,j,k</td>
</tr>
<tr>
<td>D. Ability to clearly communicate technical concepts both orally and in</td>
<td>e,f,g,h</td>
</tr>
<tr>
<td>writing</td>
<td></td>
</tr>
<tr>
<td>E. Ability to work with other disciplines</td>
<td>e,f,g,h</td>
</tr>
<tr>
<td>F. Appropriate, occasional innovation of our curriculum so it incorporates</td>
<td>l</td>
</tr>
<tr>
<td>ever-changing Computer Science technology.</td>
<td></td>
</tr>
</tbody>
</table>

5. Explain how completion of your program enables (a)-(i) of the general criteria, as well as enables the corresponding additions from the relevant program criteria.

Completion of our program ensures that a,b,c,i,j,k are met because of the core focus on the fundamentals of Computer Science. Communication skills (f) are taught via communication components in several of our required courses as well as our additional requirement, which mandates that students take either Public Speaking or Technical Writing. Teamwork skills (d) are taught in several required courses via group projects and two required Interprofessional Project (IPRO) courses. We teach ethics in the IPRO courses and we have a required senior level course that focuses on ethical issues, the need for professional development, and the impact of computing on individuals and society (e,g,h). No specific course ensures that students will be eligible for a top tier graduate program (l). However, we believe completion of the program will achieve this goal given the fact we make this standard clear to all course managers.
4. Continuous Improvement

Criterion

The program uses a documented process incorporating relevant data to regularly assess its program educational objectives and program outcomes, and to evaluate the extent to which they are being met. The results of the evaluations are documented and used to effect continuous improvement of the program through a documented plan.

1. Describe your procedure for periodically assessing the extent to which each of the program educational objectives is being met by your program. Include:
   - Frequency and timing of assessments.
   - What data are collected (should include information on initial student placement and subsequent professional development).
   - How data are collected.
   - From whom data are collected (should include students and computing professionals).
   - How assessment results are used and by whom.

We have described eight core measurement tools in Sections 2 and 3. Additionally, we have described their mapping to objectives and outcomes. Here we describe the details of core measurement tools 5, 6, and 7 used by the department to assess objectives. Subsequently, we will describe 1, 2, 3, 4, and 8 as these are used to assess outcomes. Co-Op/Employer Surveys will be an added measurement tool starting Fall 2008. At present, we rely on less formal, anecdotal conversations with employers and industry partners.

5. Recent Alumni Survey - Our department surveys Alumni periodically. In particular, we send surveys to graduating students. We also talk to alumni every semester at our ACM Employment Seminars to obtain their feedback on the program.
   a. Frequency: Every other summer.
   b. Data: The survey includes questions directly related to Computer Science Program Objectives. Responses are multiple choice or short answer.
   c. Collection Method: Email surveys.
   d. Subjects: All recent alumni (last 5 years)
   e. Usage: Alumni surveys provide an insight on the long-term fitness of the Curriculum. Consistent responses to any particular question results in large-scale reviews of the Curriculum for ability to achieve educational objectives (as they are less likely to comment on particular courses). The Undergraduate Studies Committee is in charge of identifying problems and designing and enacting solutions.

6. ACM Student Feedback Forum – Periodically (usually once per year), the CS department sponsors an ACM Student Feedback Forum. Students are encouraged to give direct feedback on any portion of the CS program.
a. Frequency: Usually once per year
b. Data: We usually give an outline of topics (Courses, Faculty, Curriculum, Labs, Servers, Software, etc.), but the student can discuss and document any issue. We request that the moderators ask students for more than just complaints, but also possible solutions.
c. Collection Method: The forum is conducted annually solely by the student members of the ACM. No faculty members are allowed in these sessions to encourage candor. One student is designated as a scribe to record comments.
d. Subjects: All undergraduate CS majors are invited.
e. Usage: The Undergraduate Studies Committee studies student comments. Comments are analyzed by the Undergraduate Committee and forwarded to the appropriate IIT entities (e.g., comments regarding building conditions are sent to the Facilities office). Comments related to the curriculum are handled by the Undergraduate Studies Committee and may lead to requests for modifications within courses.

7. IPRO Skills Assessments – The IPRO office conducts Skills Assessments throughout each semester to evaluate student teamwork, project management, applied ethics and communication skills on each IPRO project.
   a. Frequency: Every Semester
   b. Data: Likert-style questions
   c. Collection Method: The IPRO office conducts pre and post assessments to measure the level of acquisition of knowledge and skills embodied in the IPRO course objectives. Also, the IPRO office organizes workshops each semester in teamwork, project management, applied ethics and communication.
   d. Subjects: All students taking each IPRO.
   e. Usage: Periodically, usually once every other year, the IPRO office reports summary results to the Faculty.

2. Describe your procedure for periodically assessing the extent to which each of the program outcomes is being met by your program. Include:
   - Frequency and timing of assessments.
   - What data are collected (should include information on initial student placement and subsequent professional development).
   - How data are collected.
   - From whom data are collected (should include students and computing professionals).
   - How assessment results are used and by whom.
The precise mechanisms were introduced in Sections 2 and 3, as well as their mappings to particular objectives and outcomes. Here we describe the details of core measurement tools 1, 2, 3, 4, and 8 used by the department to mainly assess outcomes.

1. **Student Course Assessment (end of every semester)** - Student course surveys are given every semester by the Undergraduate Studies Committee. They are geared specifically toward asking students how well the department is meeting objectives for each course. It is our belief that if we meet objectives for each course, that the overall departmental objectives will be met.

   a. **Frequency:** Every semester
   b. **Data:** Short answer questions that directly address course objectives
   c. **Collection Method:** Anonymous paper forms collected by the Undergraduate Studies Committee
   d. **Subjects:** All students taking CS courses.
   e. **Usage:** The Undergraduate Studies Committee studies the response to each question to see how well each course is meeting its objectives. Poor performance results in an assessment of weak points in a course or curriculum, which potentially results in curricular changes, proposed by the Undergraduate Studies Committee.

2. **Student IIT evaluations** - Students are asked to fill out an evaluation by the university at the end of each semester that essentially provides direct feedback on the quality of the professor who taught the course. Clearly, the Undergraduate Studies Committee reviews these results to ensure that the quality of teaching is at a high level.

   a. **Frequency:** Every semester
   b. **Data:** Likert-style questions and short answer focused on course delivery.
   c. **Collection Method:** Anonymous online surveys collected by the University.
   d. **Subjects:** All students taking CS courses.
   e. **Usage:** The Undergraduate Studies Committee and the college study the course evaluations. The Undergraduate Studies Committee and college use these evaluations to see how well instructors are delivering their course. Poor performance results in an assessment of instructor quality, design of the course, or preparation of the student. Poorly designed courses are redesigned based on instructor, student, and other stakeholder (e.g., industry representatives) input. If student preparation is judged to be a problem, prerequisite structure changes may be made. The Undergraduate Studies Committee is in charge of identifying problems and implementing solutions.

3. **Instructor Course Assessment (end of every semester)** - Faculty are surveyed at the completion of each course that is delivered to obtain their views on whether or not the course they taught met the core objectives of the course. It is also their opportunity to suggest changes for future renditions of the course.

   a. **Frequency:** Every semester
b. Data: Short answer questions that directly address Course Objectives
c. Collection Method: Paper forms collected by the CS Undergrad Committee.
d. Subjects: All Undergraduate CS Course Instructors
e. Usage: The Undergraduate Studies Committee and Course Managers review survey responses and recommend changes to course objectives, content or delivery as needed.

4. CS Undergraduate Committee Detailed Course Review (one course every semester) –
The CS Undergraduate Committee selects one required CS course each semester for a detailed review including all assignments, lecture materials, projects, and exams. This selection is based on recent student/faculty surveys.
   a. Frequency: One course is reviewed each semester.
   b. Data: The review process involves collecting all faculty notes, student exams and student assignments and comparing them to the course objectives.
   d. Subjects: One CS course per semester.
   e. Usage: All aspects of a course are thoroughly examined to ensure they meet quality standards. If the material does not appear meet the course objectives, the instructor and Course Manager are notified and the problem is addressed.

8. The Communication Across the Curriculum (CAC) Office conducts a periodic review of communication intensive courses in each major and provides a report to each department. Communication intensive courses are identified in the Undergraduate Bulletin using a “C” designation. We use this report to evaluate the effectiveness of a student communication skills
   a. Frequency: Once every five years.
   b. Data: The review process involves collecting recent communication assignments and sample student work.
   c. Collection Method: Soft and hard copies of all communications assignments.
   d. Subjects: All undergraduate “C” courses.
   e. Usage: The CAC Office reviews the assignments and sample student materials and makes recommendations to the department for improvement in the breadth of communication skills evaluated in the “C” courses. The office also provides specific feedback on the “C” course communication assignments. The CAC office has resources to help faculty design and evaluate better communication assignments. Also, the CAC assists students by reviewing draft versions of these assignments.

3. If you have an assessment plan or similar document that provides the information in (1) and (2) above, include it as an appendix and reference the appendix here.

4. Attach as an appendix copies of the actual documentation that was used by your data collection and assessment process since the last accreditation visit or for the past three years if this is the first visit. Include survey instruments, data summaries, analysis results, etc. Indicate the appendix reference here.


5. Describe your use of the results of the program’s assessments to identify program improvements and modifications to program educational objectives and program outcomes. Include:
   - Any program changes within the last six years based on assessments.
   - Any significant future program improvement plans based upon recent assessments, including timelines.
   - Any changes in program educational objectives or program outcomes within the last six years.
   - How this information has been documented.

Any approved changes to the objectives or outcomes are posted to the Computer Science Department WWW page. If these changes require any updates to courses or program curriculum, Educational Services is also notified as they maintain the current version of the Undergraduate Catalog. See www.cs.iit.edu/~abet for supporting documentation, specifically see “Computer Science Undergraduate Studies Committee Notes”

No significant changes to the educational objectives and program outcomes have been made in the past six years.

Program changes include:

- **Fall 2001**
  - Removal of CS101 (this reduces total number of credit hours to 127)
  - Allow MATH230 as a substitute for CS330 in CS and CIS degrees
- **Spring 2002**
  - Added new CS elective course in Information Retrieval (CS429)
- **Fall 2002**
  - Change of Science/Engineering elective to Science Elective
  - Replace Non-Tech Elective in CS degree and one Free Elective in CIS degree with COM421-Technical Communications OR COM428-Verbal & Visual Communications
- **Spring 2003**
  - Added new CS elective course in Data Mining (CS422)
- **Fall 2004**
  - Change CS105/CS106 to CS115/116 (Separating majors from non-majors in the introductory programming sequence, making Java our standard introductory programming language for CS majors).
- Added new CS elective course in Information Security (CS458)
- Allowing MATH475 as option for MATH474
- Addition of Linear Algebra requirement - MATH332 or MATH333
- Restriction of Math Elective to MATH252, 410, 453, 454, 476, 482
- Removal of PHIL374 as option for CS485
- Removal of one Humanities Elective, Removal of “one social science elective must be an economics”, Humanities/Social Science requirements are now the same as the General Education Requirements
- Added 5 year BS/MS in CS

- Spring 2005
  - Added new CS elective course in Language Understanding (CS481)

- Fall 2005
  - Added two specializations (4 courses required), Specialization in Information Security, Specialization in Information and Knowledge Management Systems

- Spring 2006
  - Added new CS elective course in Information and Knowledge Management Systems (CS482)

- Fall 2006
  - Lengthen CS331, CS350, and CS351 weekly lecture from two 50-minute lectures to two 75-minute lectures. Shorten labs from one 100-minute lab to one 50-minute lab.
  - Add a required CS330 weekly recitation with TA leading.

Future Program Improvement Plans include:

- Fall 2008
  - Split CS100 into Standard/Advanced Sections
- Investigate different approaches/languages for teaching CS1 (Introductory Programming Sequence)
5. Curriculum

**Note:** One year of study refers to the amount of course work that a student would complete in an average year of fulltime enrollment. For a traditional four-year program using standard semester units, one year refers to 30 semester credits. For programs using standard quarter units, one year refers to 45 quarter credits. One year is measured similarly in programs using other units to measure course work.

**Criterion**
The program’s requirements are consistent with its educational objectives and are designed in such a way that each of the program outcomes can be achieved. The curriculum combines technical and professional requirements with general education requirements and electives to prepare students for a professional career and further study in the computing discipline associated with the program, and for functioning in modern society. The technical and professional requirements include at least one year of up-to-date coverage of basic and advanced topics in the computing discipline associated with the program. In addition, the program includes mathematics appropriate to the discipline beyond the pre-calculus level. For each course in the major required of all students, its content, expected performance criteria, and place in the overall program of study are published.

For Computer Science Programs:
Students have the following amounts of course work or equivalent educational experience.

a. Computer science: One and one-third years that includes:
   1. coverage of the fundamentals of algorithms, data structures, software design, concepts of programming languages and computer organization and architecture. [CS]
   2. an exposure to a variety of programming languages and systems. [CS]
   3. proficiency in at least one higher-level language. [CS]
   4. advanced course work that builds on the fundamental course work to provide depth. [CS]

b. One year of science and mathematics:
   1. Mathematics: At least one half year that must include discrete mathematics. The additional mathematics might consist of courses in areas such as calculus, linear algebra, numerical methods, probability, statistics, number theory, geometry or symbolic logic. [CS]
   2. Science: A science component that develops an understanding of the scientific method and provides students with an opportunity to experience this mode of inquiry in courses for science and engineering majors that provide some exposure to laboratory work. [CS]
Credit Hour Definition

One semester or quarter hour normally means one hour of lecture or three hours of laboratory per week. One academic year normally represents from twenty-eight to thirty weeks of classes, exclusive of final examinations. Please describe below if your definitions differ from these.

| One semester hour represents 50 minutes of lecture or 100-150 minutes of laboratory per week. One academic semester year entails thirty weeks of classes, exclusive of final examinations. |

A. Prerequisite Flow Chart

Attach a flow chart showing the prerequisite structure of the program’s courses required or allowed towards the major.
B. Course Requirements of Curriculum (term by term and year by year)

1. Required and elective courses. In the tables on the following pages, list the courses in the order in which they are normally taken in the curriculum, classified in the appropriate categories. The data should clearly indicate how the program meets the Curriculum Category of the *Criteria for Accrediting Computing Programs*. These tables are designed for a semester calendar; they may be easily altered for a quarter calendar.

2. Individual courses may be split between or among curriculum areas if the course content justifies the split. For example, a discrete mathematics course may have some of its semester hours under mathematics and some under computer science. In such cases, assign semester hours to categories in multiples of one-half semester hour.

3. Required courses. List courses by department/subject abbreviation (Math, Chem, IS, etc.), number, and title. Apportion the semester hours for each course by category.

4. Elective courses. Designate these courses “elective.” If an elective is restricted to a particular category, then tabulate the semester hours in that category and indicate the category in the listing, e.g. “elective—science.” In addition, be sure that you have supplied information elsewhere in this document indicating how you ensure that students take the course in the specified category (e.g., advisement, graduation check sheets, etc.). For free electives (i.e., those not restricted to a particular category), list the semester hours under the heading “Other.” Use footnotes for any listings that require further elaboration.
<table>
<thead>
<tr>
<th>Semester / Year</th>
<th>Course (Dept., Number, Title)</th>
<th>Computer Science Core</th>
<th>Computer Science Advanced</th>
<th>Math</th>
<th>Science</th>
<th>General Education</th>
<th>Other</th>
</tr>
</thead>
<tbody>
<tr>
<td>Freshman Year</td>
<td>CS100 Introduction to Professions (C)</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>CS115 Object-Oriented Programming I</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>MATH151 Calculus I</td>
<td></td>
<td></td>
<td>5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Humanities 1xx elective</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Social Science elective</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Freshman Year</td>
<td>CS116 Object-Oriented Programming II</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>CS330 Discrete Structures</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>MATH152 Calculus II</td>
<td></td>
<td></td>
<td>5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>PHYS123 Mechanics</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Humanities elective</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Freshman Year</td>
<td>CS331 Data Structures and Algorithms</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>CS350 Computer Organization and Assembly Language Programming (C)</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sophomore Year</td>
<td>MATH251 Multivariate and Vector Calculus</td>
<td></td>
<td></td>
<td>4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>PHYS221 Electromagnetism and Optics</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Social Science elective</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sophomore Year</td>
<td>CS351 Systems Programming</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>CS430 Introduction to Algorithms (C)</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>MATH332 Matrices or MATH333 Matrices and Complex Variables</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Science elective</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Humanities elective</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SUBTOTALS</td>
<td></td>
<td>18</td>
<td>3</td>
<td>17</td>
<td>11</td>
<td>15</td>
<td>0</td>
</tr>
<tr>
<td>Year Semester</td>
<td>Course (Dept., Number, Title)</td>
<td>Category (credit hours)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---------------</td>
<td>--------------------------------</td>
<td>-------------------------</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>First Semester</td>
<td>CS440 Prog. Lang. and Translators</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>CS elective (C)</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Junior Year</td>
<td>MATH474 Probability and Statistics or MATH475 Probability</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Social Science elective</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>COM421 Technical Writing or COM428 Verbal and Visual Comm.</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Second Semester</td>
<td>CS450 Operating Systems</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Math elective</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Junior Year</td>
<td>InterProfessional project (IPRO) I</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Free elective</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>First Semester</td>
<td>CS487 Software Engineering (C)</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Senior Year</td>
<td>Science elective</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>InterProfessional project (IPRO) II</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Humanities or Soc. Science elective</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Free elective</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Second Semester</td>
<td>CS485 Computers in Society (C)</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Senior Year</td>
<td>CS elective</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>CS elective</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Free elective</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Free elective</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SUBTOTALS</td>
<td></td>
<td>0 27 6 3 12 15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TOTALS</td>
<td></td>
<td>18 30 23 14 27 15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Criteria Stated Minimums</td>
<td></td>
<td>16 16 15 12 - -</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Criteria Stated Minimums</td>
<td></td>
<td>40 30 30</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

V - 6
Communication Courses - These Computer Science courses contain a communication component. As required by the university, each department is required to have 15 credit hours in major courses that are “C” courses.

Explain how the curriculum addresses the program outcomes. Include a table showing how each course contributes to the program outcomes.

<table>
<thead>
<tr>
<th>Outcome</th>
<th>Courses Supporting (ALL REQUIRED COURSES)</th>
</tr>
</thead>
</table>
| a. An ability to apply knowledge of computing and mathematics appropriate to the discipline | CS100 Introduction to the Profession (CS115/CS116 Object-Oriented Programming I & II or CS201 Accelerated Introduction to CS) (JAVA)  
CS330 Discrete Structures  
CS331 Data Structures and Algorithms  
CS350 Computer Organization  
CS351 Systems Programming  
CS430 Introduction to Algorithms  
CS440 Programming Languages and Translators  
CS450 Introduction to Operating Systems  
IPRO497 Interprofessional Projects(2) |
| b. An ability to analyze a problem, and identify and define the computing requirements appropriate to its solution | CS100 Introduction to the Profession (CS115/CS116 Object-Oriented Programming I & II or CS201 Accelerated Introduction to CS) (JAVA)  
CS330 Discrete Structures  
CS331 Data Structures and Algorithms  
CS350 Computer Organization  
CS351 Systems Programming  
CS430 Introduction to Algorithms  
CS487 Software Engineering  
IPRO497 Interprofessional Projects(2) |
| c. An ability to design, implement and evaluate a computer-based system, process, component, or program to meet desired needs | CS100 Introduction to the Profession (CS115/CS116 Object-Oriented Programming I & II or CS201 Accelerated Introduction to CS) (JAVA)  
CS331 Data Structures and Algorithms  
CS350 Computer Organization  
CS351 Systems Programming  
CS430 Introduction to Algorithms  
CS440 Programming Languages and Translators  
CS487 Software Engineering  
IPRO497 Interprofessional Projects(2) |
| d. An ability to function effectively on teams to                      | CS100 Introduction to the Profession                                                                   |
| accomplish a common goal | CS331 Data Structures and Algorithms  
CS350 Computer Organization  
CS485 Computers and Society  
CS487 Software Engineering  
IPRO497 Interprofessional Projects(2) |
|---|---|
| e. An understanding of professional, ethical, legal, security, and social issues and responsibilities | CS100 Introduction to the Profession  
CS485 Computers and Society  
CS487 Software Engineering  
IPRO497 Interprofessional Projects(2) |
| f. An ability to communicate effectively with a range of audiences | CS100 Introduction to the Profession  
CS350 Computer Organization  
CS351 Systems Programming  
CS430 Introduction to Algorithms  
CS485 Computers and Society  
CS487 Software Engineering  
General Education Humanities and Social Science Electives (7)  
COM421 Technical Communication or  
COM428 Verbal and Visual Communication  
IPRO497 Interprofessional Projects(2) |
| g. An ability to analyze the local and global impact of computing on individuals, organizations and society | CS100 Introduction to the Profession  
CS485 Computers and Society  
IPRO497 Interprofessional Projects(2) |
| h. Recognition of the need for, and an ability to engage in, continuing professional development | CS430 Introduction to Algorithms  
CS440 Programming Languages and Translators  
CS450 Introduction to Operating Systems  
CS487 Software Engineering  
IPRO497 Interprofessional Projects(2) |
| i. An ability to use current techniques, skills, and tools necessary for computing practices. | (CS115/CS116 Object-Oriented Programming I & II or CS201 Accelerated Introduction to CS) (JAVA)  
CS331 Data Structures and Algorithms  
CS350 Computer Organization  
CS351 Systems Programming  
CS430 Introduction to Algorithms  
CS440 Programming Languages and Translators  
CS487 Software Engineering |
| j. An ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices | (CS115/CS116 Object-Oriented Programming I & II or CS201 Accelerated Introduction to CS) (JAVA)  
CS330 Discrete Structures  
CS331 Data Structures and Algorithms  
CS350 Computer Organization  
CS351 Systems Programming  
CS411 Computer Graphics |
k. An ability to apply design and development principles in the construction of software systems of varying complexity (CS115/CS116 Object-Oriented Programming I & II or CS201 Accelerated Introduction to CS) (JAVA) CS331 Data Structures and Algorithms CS487 Software Engineering

l. Be prepared to enter a top-ranked graduate program in Computer Science. CS430 Introduction to Algorithms CS440 Programming Languages and Translators CS450 Introduction to Operating Systems CS487 Software Engineering

---

**For computer science programs**

The following areas must be stressed within the program’s curriculum. Indicate the course numbers and titles of courses embodying a significant portion of these areas:

<table>
<thead>
<tr>
<th>Area</th>
<th>Criterion</th>
<th>Courses (Dept., Number, and Title) (ALL REQUIRED COURSES)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithms</td>
<td>a.1.</td>
<td>CS330 Discrete Structures CS331 Data Structures and Algorithms CS430 Introduction to Algorithms</td>
</tr>
<tr>
<td>Data structures</td>
<td>a.1.</td>
<td>CS330 Discrete Structures CS331 Data Structures and Algorithms CS430 Introduction to Algorithms</td>
</tr>
<tr>
<td>Software design</td>
<td>a.1.</td>
<td>CS115/CS116 Object-Oriented Programming I &amp; II or CS201 Accelerated Introduction to CS CS331 Data Structures and Algorithms CS487 Software Engineering</td>
</tr>
<tr>
<td>Programming language concepts</td>
<td>a.1.</td>
<td>(CS115/CS116 Object-Oriented Programming I &amp; II or CS201 Accelerated Introduction to CS) CS331 Data Structures and Algorithms CS440 Programming languages and Translators</td>
</tr>
<tr>
<td>Computer organization and architecture</td>
<td>a.1.</td>
<td>CS350 Computer Organization and Assembly Language Programming CS351 Systems Programming</td>
</tr>
<tr>
<td>Exposure to variety of languages and systems</td>
<td>a.2.</td>
<td>(CS115/CS116 Object-Oriented Programming I &amp; II or CS201 Accelerated Introduction to CS) (JAVA) CS350 Computer Organization and Assembly Language Programming (C/C++) CS351 Systems Programming (C/C++) CS440 Programming Languages and Translators (Ocaml, Prolog, Scheme, Forth, and APL)</td>
</tr>
</tbody>
</table>
| Proficiency in at least one higher level language | a.3. | CS331 Data Structures and Algorithms (JAVA)  
| | | CS351 Systems Programming (C/C++) |
| Advanced course work that builds on the fundamental course work to provide depth | a.4. | CS430 Introduction to Algorithms  
| | | CS440 Programming languages and Translators  
| | | CS450 Operating Systems  
| | | CS487 Software Engineering  
| | | CS4xx/5xx Computer Science Electives (5) |
| Discrete mathematics | b.1. | CS330 Discrete Structures |
| Science component | b.2. | PHYS123 Mechanics  
| | | PHYS221 Electromagnetism and Optics  
| | | Science Electives (2) |

**C. Course Descriptions**

For each required or elective course in the program that can be counted in the curriculum being reviewed for accreditation, include a two-page or three-page course outline, as indicated below, at this point in the Self-Study Report. If your documentation does not exactly follow this format, be sure that all of the requested information (if applicable) is present, and please in any case adhere to a common format for all course descriptions. If some of this documentation is on-line (e.g., in an instructor’s web site), please give here the URLs for accessing any such materials. These URLs should be made accessible to the visiting team as soon as the Self-Study is sent to them.

As described in Section H of the General Instructions for the Self-Study, the course outline for each required or elective computing course in the program (including those that satisfy the IS Environment component of an Information Systems program) must also be included in a display of course materials that is available for study at all times during the evaluation visit.
Required Computer Science Courses
COURSE DESCRIPTION

<table>
<thead>
<tr>
<th>Dept., Number</th>
<th>Course Number</th>
<th>Course Title</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>CS100</td>
<td>Introduction to the Profession</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Semester hours</th>
<th>Course Coordinator</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>Matthew Bauer, Senior Lecturer</td>
</tr>
</tbody>
</table>

Current Catalog Description

An introduction to science and engineering as a profession. Examines the problem-solving process used in engineering and science. Emphasizes the interdisciplinary and international nature of problem solving and the need to evaluate solutions in terms of a variety of constraints: computational, financial, and social. (1-2-2) (C)

Textbook

- Online Readings and Demonstrations – see www.cs.iit.edu/~cs100
- or
- Structure and Interpretation of Computer Programs, by Abelson, Sussman, and Sussman, 1984

References


Course Outcomes

Students should be able to:
- Demonstrate a basic understanding of these computer science concepts: Algorithms, Artificial Intelligence, Data Structures, Databases, Ethics/Society/History, Machine Architecture, Networking, Number Systems, Operating Systems, Programming Languages, Software Engineering
- Demonstrate basic problem solving, sorting and recursion using pseudocode.
- Apply the ACM Ethics Code to a real world computer science related scenario.
- Work in small teams on Computer Science deliverables.
- Communicate Computer Science deliverables.

Relationship between Course Outcomes and Program Outcomes

The following Program Outcomes are supported by the above Course Outcomes:
- a. An ability to apply knowledge of computing and mathematics appropriate to the discipline
- b. An ability to analyze a problem, and identify and define the computing requirements appropriate to its solution
- c. An ability to design, implement and evaluate a computer-based system, process, component, or program to meet desired needs
• d. An ability to function effectively on teams to accomplish a common goal
• e. An understanding of professional, ethical, legal, security, and social issues and responsibilities
• f. An ability to communicate effectively with a range of audiences
• g. An ability to analyze the local and global impact of computing on individuals, organizations and society

Prerequisites by Topic
no prerequisites

Major Topics Covered in the Course

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Goals</td>
<td>1 hour</td>
</tr>
<tr>
<td>2. History Of Computing</td>
<td>1 hour</td>
</tr>
<tr>
<td>3. Pseudocode &amp; Problem Solving</td>
<td>2 hours</td>
</tr>
<tr>
<td>4. Binary Arithmetic Program Translation Logic</td>
<td>3 hours</td>
</tr>
<tr>
<td>5. Library Research - Cutting Edge Technologies</td>
<td>2 hours</td>
</tr>
<tr>
<td>6. Career Development Center – Resume; Counseling and Health Services</td>
<td>2 hours</td>
</tr>
<tr>
<td>7. PC Architecture</td>
<td>1 hour</td>
</tr>
<tr>
<td>8. Pseudocode &amp; Sorting</td>
<td>2 hour</td>
</tr>
<tr>
<td>9. Operating Systems</td>
<td>1 hour</td>
</tr>
<tr>
<td>10. Ethics</td>
<td>1 hour</td>
</tr>
<tr>
<td>11. RobotWars</td>
<td>2 hours</td>
</tr>
<tr>
<td>12. Pseudocode &amp; Recursion</td>
<td>2 hours</td>
</tr>
<tr>
<td>13. The Internet - Theory &amp; Practice</td>
<td>5 hours</td>
</tr>
<tr>
<td>14. Presentation Skills</td>
<td>4 hours</td>
</tr>
<tr>
<td>Midterm Exam</td>
<td>1 hour</td>
</tr>
<tr>
<td>Final Exam</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>30 hours</td>
</tr>
</tbody>
</table>

Assessment Plan for the Course

End of every semester Course Objective Assessments by CS department. End of semester Course Evaluations by IIT. Reviewed every Spring semester by CS Undergraduate Studies Committee for possible updates in the following Fall. Once every 4-5 years a detailed review of all materials for the course is made by the CS Undergraduate Studies Committee.

How Data in the Course is Used to Assess Program Outcomes (unless adequately covered already in the assessment discussion under Criterion 4)

See the assessment discussion under Criterion 4

For a computer science program
Estimate Curriculum Category Content (Semester hours)
<table>
<thead>
<tr>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithms</td>
<td>.5</td>
<td></td>
<td>Software design</td>
<td>.5</td>
<td></td>
</tr>
<tr>
<td>Data structures</td>
<td>.5</td>
<td></td>
<td>Concepts of programming languages</td>
<td>.5</td>
<td></td>
</tr>
</tbody>
</table>
COURSE DESCRIPTION

<table>
<thead>
<tr>
<th>Dept., Number</th>
<th>Course Title</th>
<th>Introduction to Computer Programming I</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS105</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Semester hours | 2 | Course Coordinator | Matthew Bauer, Senior Lecturer |

Current Catalog Description
Introduces the use of a high-level programming language (C/C++) as a problem-solving tool—including basic data structures and algorithms, structured programming techniques, and software documentation. Designed for students who have had little or no prior experience with computer programming. (2-1-2) NOT FOR COMPUTER SCIENCE MAJORS, SERVICE COURSE

Textbook
or
Learn to Program, Pine, 2006

References
none

Course Outcomes
Students should be able to:

- Analyze and explain the behavior of simple programs involving the following fundamental programming constructs: assignment, I/O (including file I/O), selection, iteration, functions
- Write a program that uses each of the following fundamental programming constructs: assignment, I/O (including file I/O), selection, iteration, functions
- Break a problem into logical pieces that can be solved (programmed) independently.
- Develop, and analyze, algorithms for solving simple problems.
- Use a suitable programming language, and development environment, to implement, test, and debug algorithms for solving simple problems.
- Write programs that use each of the following data structures (and describe how they are represented in memory): strings, arrays, and class libraries including strings and vectors

Relationship between Course Outcomes and Program Outcomes
NOT FOR COMPUTER SCIENCE MAJORS, SERVICE COURSE

Prerequisites by Topic
no prerequisites

Major Topics Covered in the Course
1. Development Environment, C++ Program Elements 3 hours
2. Data Types, Expressions, Basic I/O, Data Type Conversion, Library Functions, Strings (introduction) 3 hours
3. Selection 6 hours
4. Stream File I/O, Output Manipulators 4 hours
5. Iteration 8 hours
6. Functions (scope, pass by reference, overloading) 3 hours
7. Arrays, Vector Class 9 hours
8. Project 5 hours
Quiz #1, Midterm Exam, Quiz #2 4 hours
Final Exam 45 hours

Assessment Plan for the Course
End of every semester Course Objective Assessments by CS department. End of semester Course Evaluations by IIT. Reviewed every Spring semester by CS Undergraduate Studies Committee for possible updates in the following Fall. Once every 4-5 years a detailed review of all materials for the course is made by the CS Undergraduate Studies Committee.

How Data in the Course is Used to Assess Program Outcomes (unless adequately covered already in the assessment discussion under Criterion 4)
See the assessment discussion under Criterion 4

For a computer science program
Estimate Curriculum Category Content (Semester hours)
NOT FOR COMPUTER SCIENCE MAJORS, SERVICE COURSE
COURSE DESCRIPTION

<table>
<thead>
<tr>
<th>Dept., Number</th>
<th>Course Title</th>
<th>Object-Oriented Programming I</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS115</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Semester hours 2 Course Coordinator Matthew Bauer, Senior Lecturer

Current Catalog Description
Introduces the use of a high-level object-oriented programming language as a problem-solving tool – including basic data structures and algorithms, object-oriented programming techniques, and software documentation. Designed for students who have had little or no prior experience with computer programming. For students in CS and CS related degree programs. (2-1-2)

Textbook

References
none

Course Outcomes
Students should be able to:
- Analyze and explain the behavior of simple programs involving the following fundamental programming constructs: assignment, I/O (including file I/O), selection, iteration, methods
- Write a program that uses each of the following fundamental programming constructs: assignment, I/O (including file I/O), selection, iteration, methods
- Break a problem into logical pieces that can be solved (programmed) independently.
- Develop, and analyze, algorithms for solving simple problems.
- Use a suitable programming language, and development environment, to implement, test, and debug algorithms for solving simple problems.
- Write programs that use each of the following data structures (and describe how they are represented in memory): strings, arrays
- Explain and apply object-oriented design and testing involving the following concepts: data abstraction, encapsulation, information hiding
- Use a development environment to design, code, test, and debug simple programs, including multi-file source projects, in an object-oriented programming language.
- Implement basic error handling
- Apply appropriate problem-solving strategies
- Use APIs (Application Programmer Interfaces) and design/program APIs

Relationship between Course Outcomes and Program Outcomes
The following Program Outcomes are supported by the above Course Outcomes:

- a. An ability to apply knowledge of computing and mathematics appropriate to the discipline
- b. An ability to analyze a problem, and identify and define the computing requirements appropriate to its solution
- c. An ability to design, implement and evaluate a computer-based system, process, component, or program to meet desired needs
- i. An ability to use current techniques, skills, and tools necessary for computing practices
- j. An ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices
- k. An ability to apply design and development principles in the construction of software systems of varying complexity

Prerequisites by Topic

no prerequisites

Major Topics Covered in the Course

<table>
<thead>
<tr>
<th>1. Fundamental data storage and manipulation (types and variables, statements and expressions)</th>
<th>6 hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>2. Functions</td>
<td>4 hours</td>
</tr>
<tr>
<td>3. Classes (classes and objects, instance variables and instance methods, and encapsulation)</td>
<td>6 hours</td>
</tr>
<tr>
<td>4. Flow of control (Boolean expressions, conditional statements, and loops)</td>
<td>12 hours</td>
</tr>
<tr>
<td>5. Vectors</td>
<td>6 hours</td>
</tr>
<tr>
<td>6. Problem Solving approaches (This section is dispersed appropriately throughout the semester to illustrate the above techniques.)</td>
<td>3 hours</td>
</tr>
<tr>
<td>7. Software Engineering – design, testing, debugging (This section is dispersed appropriately throughout the semester to illustrate the above techniques.)</td>
<td>6 hours</td>
</tr>
<tr>
<td>Exams</td>
<td>2 hours</td>
</tr>
<tr>
<td>Final Exam</td>
<td>-</td>
</tr>
</tbody>
</table>

Exams 2 hours

Final Exam 45 hours

Assessment Plan for the Course

End of every semester Course Objective Assessments by CS department. End of semester Course Evaluations by IIT. Reviewed every Spring semester by CS Undergraduate Studies Committee for possible updates in the following Fall. Once every 4-5 years a detailed review of all materials for the course is made by the CS Undergraduate Studies Committee.

How Data in the Course is Used to Assess Program Outcomes (unless adequately covered already in the assessment discussion under Criterion 4)

See the assessment discussion under Criterion 4

For a computer science program
## Estimate Curriculum Category Content (Semester hours)

<table>
<thead>
<tr>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithms</td>
<td>.33</td>
<td></td>
<td>Software design</td>
<td>.33</td>
<td></td>
</tr>
<tr>
<td>Data structures</td>
<td>.33</td>
<td></td>
<td>Concepts of programming languages</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>
# COURSE DESCRIPTION

<table>
<thead>
<tr>
<th>Dept., Number</th>
<th>Course Title</th>
<th>Object-Oriented Programming II</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS116</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Semester hours</th>
<th>Course Coordinator</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>Matthew Bauer, Senior Lecturer</td>
</tr>
</tbody>
</table>

**Current Catalog Description**

Continuation of CS 115. Introduces more advanced elements of object-oriented programming – including dynamic data structures, recursion, searching and sorting, and advanced object-oriented programming techniques. For students in CS and CS related degree programs.

Prerequisite: CS 115 (2-1-2)

**Textbook**


**References**

none

**Course Outcomes**

Students should be able to:

- Analyze and explain the behavior of simple programs involving the following fundamental programming constructs: assignment, I/O (including file I/O), selection, iteration, methods
- Write a program that uses each of the following fundamental programming constructs: assignment, I/O (including file I/O), selection, iteration, methods
- Break a problem into logical pieces that can be solved (programmed) independently.
- Develop, and analyze, algorithms for solving simple problems.
- Use a suitable programming language, and development environment, to implement, test, and debug algorithms for solving simple problems.
- Write programs that use each of the following data structures (and describe how they are represented in memory): strings, arrays
- Explain the basics of the concept of recursion.
- Write, test, and debug simple recursive functions and procedures.
- Explain and apply object-oriented design and testing involving the following concepts: data abstraction, encapsulation, information hiding, inheritance, polymorphism
- Use a development environment to design, code, test, and debug simple programs, including multi-file source projects, in an object-oriented programming language.
- Implement basic error handling
- Solve problems by creating and using sequential search, binary search, and quadratic sorting algorithms (selection, insertion)
• Determine the time complexity of simple algorithms.
• Apply appropriate problem-solving strategies
• Use APIs (Application Programmer Interfaces) and design/program APIs

Relationship between Course Outcomes and Program Outcomes
The following Program Outcomes are supported by the above Course Outcomes:
• a. An ability to apply knowledge of computing and mathematics appropriate to the discipline
• b. An ability to analyze a problem, and identify and define the computing requirements appropriate to its solution
• c. An ability to design, implement and evaluate a computer-based system, process, component, or program to meet desired needs
• i. An ability to use current techniques, skills, and tools necessary for computing practices
• j. An ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices
• k. An ability to apply design and development principles in the construction of software systems of varying complexity

Prerequisites by Topic
CS115 - Basic object-oriented programming concepts

Major Topics Covered in the Course

<table>
<thead>
<tr>
<th>Topic</th>
<th>Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Review of CS115 material</td>
<td>12 hours</td>
</tr>
<tr>
<td>2. Inheritance (subclasses, dynamic binding, abstract classes, and interfaces)</td>
<td>6 hours</td>
</tr>
<tr>
<td>3. Strings</td>
<td>3 hours</td>
</tr>
<tr>
<td>4. Introduction to recursion</td>
<td>4 hours</td>
</tr>
<tr>
<td>5. Searching and sorting algorithms (linear and binary search, selection sort, insertion sort, and quick sort - introduced via recursive versions)</td>
<td>6 hours</td>
</tr>
<tr>
<td>6. Algorithm analysis</td>
<td>3 hours</td>
</tr>
<tr>
<td>7. Problem Solving approaches (This section is dispersed appropriately throughout the semester to illustrate the above techniques.)</td>
<td>3 hours</td>
</tr>
<tr>
<td>8. Software Engineering – design, testing, debugging (This section is dispersed appropriately throughout the semester to illustrate the above techniques.)</td>
<td>6 hours</td>
</tr>
<tr>
<td>Exams</td>
<td>2 hours</td>
</tr>
<tr>
<td>Final Exam</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>45 hours</td>
</tr>
</tbody>
</table>

Assessment Plan for the Course
End of every semester Course Objective Assessments by CS department. End of semester Course Evaluations by IIT. Reviewed every Spring semester by CS Undergraduate Studies Committee for possible updates in the following Fall. Once every 4-5 years a detailed review of all materials for the course is made by the CS Undergraduate Studies Committee.
How Data in the Course is Used to Assess Program Outcomes (unless adequately covered already in the assessment discussion under Criterion 4)

See the assessment discussion under Criterion 4

For a computer science program

Estimate Curriculum Category Content (Semester hours)

<table>
<thead>
<tr>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithms</td>
<td>.33</td>
<td></td>
<td>Software design</td>
<td>.33</td>
<td></td>
</tr>
<tr>
<td>Data structures</td>
<td>.33</td>
<td></td>
<td>Concepts of programming languages</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>
# COURSE DESCRIPTION

<table>
<thead>
<tr>
<th>Dept., Number</th>
<th>Course Title</th>
<th>Accelerated Introduction to Computer Science</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS201</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Semester hours</th>
<th>Course Coordinator</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>Matthew Bauer, Senior Lecturer</td>
</tr>
</tbody>
</table>

Current Catalog Description

Problem solving and design using an object-oriented programming language. Introduces a variety of problem solving techniques, algorithms, and data structures in object-oriented programming. Prerequisites: CS105 or CS 115 or experience using any programming language. (3-2-4)

Textbook


References

None

Course Outcomes

Students should be able to:

- Analyze and explain the behavior of simple programs involving the following fundamental programming constructs: assignment, I/O (including file I/O), selection, iteration, methods
- Write a program that uses each of the following fundamental programming constructs: assignment, I/O (including file I/O), selection, iteration, methods
- Break a problem into logical pieces that can be solved (programmed) independently.
- Develop, and analyze, algorithms for solving simple problems.
- Use a suitable programming language, and development environment, to implement, test, and debug algorithms for solving simple problems.
- Write programs that use each of the following data structures (and describe how they are represented in memory): strings, arrays
- Explain the basics of the concept of recursion.
- Write, test, and debug simple recursive functions and procedures.
- Explain and apply object-oriented design and testing involving the following concepts: data abstraction, encapsulation, information hiding, inheritance, polymorphism
- Use a development environment to design, code, test, and debug simple programs, including multi-file source projects, in an object-oriented programming language.
- Implement basic error handling
- Solve problems by creating and using sequential search, binary search, and quadratic sorting algorithms (selection, insertion)
• Determine the time complexity of simple algorithms.
• Apply appropriate problem-solving strategies
• Use APIs (Application Programmer Interfaces) and design/program APIs

Relationship between Course Outcomes and Program Outcomes

The following Program Outcomes are supported by the above Course Outcomes:

• a. An ability to apply knowledge of computing and mathematics appropriate to the discipline
• b. An ability to analyze a problem, and identify and define the computing requirements appropriate to its solution
• c. An ability to design, implement and evaluate a computer-based system, process, component, or program to meet desired needs
• i. An ability to use current techniques, skills, and tools necessary for computing practices
• j. An ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices
• k. An ability to apply design and development principles in the construction of software systems of varying complexity

Prerequisites by Topic

CS105 or CS 115 or experience using any programming language.

Major Topics Covered in the Course

<table>
<thead>
<tr>
<th>Topic</th>
<th>Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Fundamental data storage and manipulation (types and variables,</td>
<td>5</td>
</tr>
<tr>
<td>statements and expressions)</td>
<td></td>
</tr>
<tr>
<td>2. Functions</td>
<td>3</td>
</tr>
<tr>
<td>3. Classes (classes and objects, instance variables and instance</td>
<td>5</td>
</tr>
<tr>
<td>methods, and encapsulation)</td>
<td></td>
</tr>
<tr>
<td>4. Flow of control (Boolean expressions, conditional statements,</td>
<td>10</td>
</tr>
<tr>
<td>and loops)</td>
<td></td>
</tr>
<tr>
<td>5. Vectors</td>
<td>5</td>
</tr>
<tr>
<td>6. Review of CS115 material</td>
<td>10</td>
</tr>
<tr>
<td>7. Inheritance (subclasses, dynamic binding, abstract classes, and</td>
<td>5</td>
</tr>
<tr>
<td>interfaces)</td>
<td></td>
</tr>
<tr>
<td>8. Strings</td>
<td>3</td>
</tr>
<tr>
<td>9. Introduction to recursion.</td>
<td>3</td>
</tr>
<tr>
<td>10. Searching and sorting algorithms (linear and binary search,</td>
<td>5</td>
</tr>
<tr>
<td>selection sort, insertion sort, and quick sort - introduced via</td>
<td></td>
</tr>
<tr>
<td>recursive versions)</td>
<td></td>
</tr>
<tr>
<td>11. Algorithm analysis.</td>
<td>2</td>
</tr>
<tr>
<td>12. Problem Solving approaches (This section is dispersed</td>
<td>5</td>
</tr>
<tr>
<td>appropriately throughout the semester to illustrate</td>
<td></td>
</tr>
<tr>
<td>the above techniques.)</td>
<td></td>
</tr>
<tr>
<td>13. Software Engineering – design, testing, debugging (This section</td>
<td>10</td>
</tr>
<tr>
<td>is dispersed appropriately throughout the semester to illustrate</td>
<td></td>
</tr>
<tr>
<td>the above techniques.)</td>
<td></td>
</tr>
<tr>
<td>Exams</td>
<td>4</td>
</tr>
<tr>
<td>Final Exam</td>
<td>75</td>
</tr>
</tbody>
</table>
Assessment Plan for the Course

End of every semester Course Objective Assessments by CS department. End of semester Course Evaluations by IIT. Reviewed every Spring semester by CS Undergraduate Studies Committee for possible updates in the following Fall. Once every 4-5 years a detailed review of all materials for the course is made by the CS Undergraduate Studies Committee.

How Data in the Course is Used to Assess Program Outcomes (unless adequately covered already in the assessment discussion under Criterion 4)

See the assessment discussion under Criterion 4

For a computer science program
Estimate Curriculum Category Content (Semester hours)

<table>
<thead>
<tr>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithms</td>
<td>.66</td>
<td></td>
<td>Software design</td>
<td>.66</td>
<td></td>
</tr>
<tr>
<td>Data structures</td>
<td>.66</td>
<td></td>
<td>Concepts of programming languages</td>
<td>2</td>
<td></td>
</tr>
</tbody>
</table>
### COURSE DESCRIPTION

<table>
<thead>
<tr>
<th>Dept., Number</th>
<th>Course Title</th>
<th>Discrete Structures</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS330</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Semester hours</th>
<th>Course Coordinator</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>Dr. Sanjiv Kapoor, Professor</td>
</tr>
</tbody>
</table>

**Current Catalog Description**

Introduction to the use of formal mathematical structures to represent problems and computational processes. Topics covered include Boolean algebra, first-order logic, recursive structures, graphs, and abstract language models. Prerequisite: CS 116 or CS 201. (3-0-3)

**Textbook**


**References**

None

**Course Outcomes**

Students should be able to:

- Illustrate by examples the basic terminology of functions, relations, and sets and demonstrate knowledge of their associated operations.
- Demonstrate in practical applications the use of basic counting principles of permutations, combinations, inclusion/exclusion principle and the pigeonhole methodology.
- Calculate probabilities of events and expectations of random variables for problems arising from games of chance.
- Establish and solve recurrence relations that arise in counting problems including the problem of determining the time complexity of recursively defined algorithms.
- Model logic statements arising in algorithm correctness and real-life situations and manipulate them using the formal methods of propositional and predicate logic.
- Outline basic proofs for theorems using the techniques of - direct proofs, proof by counterexample, proof by contraposition, proof by contradiction, mathematical induction.
- Relate the ideas of mathematical induction to recursion and recursively defined structures.
- Illustrate by example basic terminology of graph theory and model problems in computer science using graphs and trees.
- Deduce properties that establish particular graphs as Trees, Planar, Eulerian, and Hamiltonian.
- Illustrate the application of trees and graphs to data structures.
- Explain the basic concepts modeling computation including formal machines, languages, finite automata, Turing machines.

**Relationship between Course Outcomes and Program Outcomes**
The following Program Outcomes are supported by the above Course Outcomes:

- a. An ability to apply knowledge of computing and mathematics appropriate to the discipline
- b. An ability to analyze a problem, and identify and define the computing requirements appropriate to its solution
- j. An ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices

Prerequisites by Topic
CS 116 or CS 201 - Experience with basic programming constructs and algorithms

Major Topics Covered in the Course

<table>
<thead>
<tr>
<th>Topic</th>
<th>Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Sets, Functions and relations - sets, set operations, functions, summations, growth of functions, equivalence relations, countable and uncountable sets, examples of algorithm analysis</td>
<td>4.5</td>
</tr>
<tr>
<td>2. Counting Methods – permutations, combinations, discrete probability, pigeonhole principle</td>
<td>6</td>
</tr>
<tr>
<td>3. Advanced counting – inclusion-exclusion, recurrence relations, methods of solving recurrences, examples from computer sciences</td>
<td>6</td>
</tr>
<tr>
<td>4. Introductory Logic – propositional logic, predicate logic, proof methodologies, examples of algorithm correctness</td>
<td>6</td>
</tr>
<tr>
<td>5. Partially Ordered sets - trees, boolean algebra, example of minimizing circuits</td>
<td>4.5</td>
</tr>
<tr>
<td>6. Introduction to Graphs - trees, connectivity, eulerian traversals, minimum spanning tree, planarity, Euler’s formula, matching</td>
<td>7.5</td>
</tr>
<tr>
<td>7. Formal machines and languages-an introduction - automaton, grammars and turing machines</td>
<td>6</td>
</tr>
<tr>
<td>8. Introduction to Algebraic Topics (OPTIONAL) – rings, groups, semi-groups.</td>
<td>1.5</td>
</tr>
<tr>
<td>Exam #1, Exam #2</td>
<td>3</td>
</tr>
<tr>
<td>Final Exam</td>
<td>-</td>
</tr>
</tbody>
</table>

Assessment Plan for the Course

End of every semester Course Objective Assessments by CS department. End of semester Course Evaluations by IIT. Reviewed every Spring semester by CS Undergraduate Studies Committee for possible updates in the following Fall. Once every 4-5 years a detailed review of all materials for the course is made by the CS Undergraduate Studies Committee.

How Data in the Course is Used to Assess Program Outcomes (unless adequately covered already in the assessment discussion under Criterion 4)

See the assessment discussion under Criterion 4

For a computer science program
<table>
<thead>
<tr>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithms</td>
<td>2</td>
<td></td>
<td>Software design</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Data structures</td>
<td>1</td>
<td></td>
<td>Concepts of programming languages</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
COURSE DESCRIPTION

<table>
<thead>
<tr>
<th>Dept., Number</th>
<th>Course Title</th>
<th>Data Structures and Algorithms</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS331</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Semester hours | 3 | Course Coordinator | Dr. Gruia Calinescu, Associate Professor |

Current Catalog Description

Implementation and application of the essential data structures used in computer science. Analysis of basic sorting and searching algorithms and their relationship to these data structures. Particular emphasis is given to the use of object-oriented design and data abstraction in the creation and application of data structures. Prerequisite: CS 116 or CS 201. (2-2-3)

Textbook

Teacher Supplied Material - http://dijkstra.cs.iit.edu/cs331-sp08/schedule/

References

http://dijkstra.cs.iit.edu/cs331-sp08/resources/

Course Outcomes

Students should be able to:

- Explain, implement, and apply the following data-structures:
  - lists (unordered and ordered), stacks, queues, expression trees, binary search trees, heaps, and hash tables.
- Analyze the time and space complexity of algorithms using asymptotic upper bounds (big-O notation).
- Explain and use references and linked structures.
- Outline basic object-oriented design concepts: composition, inheritance, and polymorphism.
- Write and test recursive procedures, and explain the run-time stack concept.
- Analyze searching and sorting algorithms, and explain their relationship to data-structures.
- Choose and implement appropriate data-structures to solve an application problem.
- Explain how to use unit tests and version control in your software development.

Relationship between Course Outcomes and Program Outcomes

The following Program Outcomes are supported by the above Course Outcomes:

- a. An ability to apply knowledge of computing and mathematics appropriate to the discipline
- b. An ability to analyze a problem, and identify and define the computing requirements appropriate to its solution
- c. An ability to design, implement and evaluate a computer-based system, process, component, or program to meet desired needs
d. An ability to function effectively on teams to accomplish a common goal
i. An ability to use current techniques, skills, and tools necessary for computing practices.
j. An ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices
k. An ability to apply design and development principles in the construction of software systems of varying complexity

Prerequisites by Topic
CS 116 or CS 201 - Experience in object-oriented programming

Major Topics Covered in the Course

| 1. Abstraction/Variables               | 3 hours |
| 2. Linux/Subversion                   | 3 hours |
| 3. Lists (Array and Linked List)      | 7.5 hours |
| 4. Stacks and Queues                  | 6 hours |
| 5. Ordered Lists, Sorting             | 7.5 hours |
| 6. Doubly-Linked Lists                | 4.5 hours |
| 7. Binary Search Trees                | 6 hours |
| 8. Expression Trees                   | 3 hours |
| 9. Heaps                               | 4.5 hours |
| 10. Hash Tables                       | 6 hours |
| 11. Project(s) discussion, Midterm(s) and discussion, Project(s) evaluation | 9 hours |
| Final Exam                            | - |
|                                      | 60 hours |

Assessment Plan for the Course

End of every semester Course Objective Assessments by CS department. End of semester Course Evaluations by IIT. Reviewed every Spring semester by CS Undergraduate Studies Committee for possible updates in the following Fall. Once every 4-5 years a detailed review of all materials for the course is made by the CS Undergraduate Studies Committee.

How Data in the Course is Used to Assess Program Outcomes (unless adequately covered already in the assessment discussion under Criterion 4)
See the assessment discussion under Criterion 4

For a computer science program

Estimate Curriculum Category Content (Semester hours)

<table>
<thead>
<tr>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithms</td>
<td>.33</td>
<td></td>
<td>Software design</td>
<td>.33</td>
<td></td>
</tr>
<tr>
<td>Data structures</td>
<td>1.5</td>
<td>.5</td>
<td>Concepts of programming languages</td>
<td>.33</td>
<td></td>
</tr>
</tbody>
</table>
COURSE DESCRIPTION

<table>
<thead>
<tr>
<th>Dept., Number</th>
<th>Course Title</th>
<th>Computer Organization and Assembly Language Programming</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS350</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Semester hours 3 Course Coordinator Dr. Cindy Hood, Associate Professor

Current Catalog Description
Introduction to the internal architecture of computer systems. Focuses on the relationship between a computer's hardware, its native instruction set, and the implementation of high-level languages on that machine. Lab exercises focused on assembly language programming and simple processor design explore and analyze computer architecture. Prerequisite: CS 116 or CS 201. (2-2-3) (C)

Textbook
Introduction to Computing Systems: From Bits and Gates to C and Beyond, 2/e; Yale N. Patt, Sanjay J. Patel, McGraw-Hill, 2004

References

Course Outcomes
The focus of this course will be how computers work with particular focus on the relationship between software written in a high level language and the computer systems that compile and execute them. Specifically, we will be uncovering details of the lower layers of abstraction present in computer systems by discussing topics which include
- Digital Logic Hardware
- Low Level Data Representation
- von Neumann Computing Model
- Instruction Set Architectures
- Assembly Language
- Input/Output
- Traps and Subroutines
- Interrupts
- Call Stack
- C programming

We will study these concepts in the context of the LC-3 processor as well as the ARM architecture. We will explore the relationship between hardware and software by programming an ARM-based device, the Nintendo Gameboy Advance in C. Students will gain an understanding of all the components of a computer, insight into the interactions between software and hardware, and an appreciation for the advantages and limitations of the abstractions provided by higher-level languages.
Relationship between Course Outcomes and Program Outcomes

The following Program Outcomes are supported by the above Course Outcomes:

- a. An ability to apply knowledge of computing and mathematics appropriate to the discipline
- b. An ability to analyze a problem, and identify and define the computing requirements appropriate to its solution
- c. An ability to design, implement and evaluate a computer-based system, process, component, or program to meet desired needs
- d. An ability to function effectively on teams to accomplish a common goal
- e. An ability to communicate effectively with a range of audiences
- f. An ability to use current techniques, skills, and tools necessary for computing practices.
- g. An ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices

Prerequisites by Topic

CS 116 or CS 201 - Experience in object-oriented programming

Major Topics Covered in the Course

1. Digital Logic Hardware 3 hours
2. Low Level Data Representation 3 hours
3. von Neumann Computing Model 6 hours
4. Instruction Set Architectures 6 hours
5. Assembly Language 6 hours
6. Input/Output 6 hours
7. Traps and Subroutines 6 hours
8. Interrupts 6 hours
9. Call Stack 6 hours
10. C programming 6 hours
11. Project(s) discussion, Midterm(s) and discussion, Project(s) evaluation 6 hours

Final Exam - 60 hours

Assessment Plan for the Course

End of every semester Course Objective Assessments by CS department. End of semester Course Evaluations by IIT. Reviewed every Spring semester by CS Undergraduate Studies Committee for possible updates in the following Fall. Once every 4-5 years a detailed review of all materials for the course is made by the CS Undergraduate Studies Committee.

How Data in the Course is Used to Assess Program Outcomes (unless adequately covered already in the assessment discussion under Criterion 4)

See the assessment discussion under Criterion 4
For a computer science program
Estimate Curriculum Category Content (Semester hours)

<table>
<thead>
<tr>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithms</td>
<td></td>
<td></td>
<td>Software design</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Data structures</td>
<td></td>
<td></td>
<td>Concepts of programming languages</td>
<td></td>
<td>3</td>
</tr>
</tbody>
</table>

V - 33
COURSE DESCRIPTION

<table>
<thead>
<tr>
<th>Dept., Number</th>
<th>Course Title</th>
<th>Systems Programming</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS351</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Semester hours</td>
<td>3</td>
<td>Matthew Bauer, Senior Lecturer</td>
</tr>
</tbody>
</table>

Current Catalog Description

Examines the components of sophisticated multi-layer software systems-including device drivers, systems software, applications interfaces, and user interfaces. Explores the design and development of interrupt-driven and event-driven software. Prerequisites: CS 331, CS 350. (2-2-3)

Textbook


References

Rochkind, Marc J. Advanced UNIX Programming. Addison-Wesley, 2004
http://www.cs.iit.edu/~lee/cs351/resources.shtml

Course Outcomes

Students should be able to:

- Define the concept and role of a process in a modern operating system
- Describe the key abstractions an operating system provides to running processes
- Describe the function, usage, and operation of system calls related to process management, memory management and I/O
- Explain exceptional control flow, including:
  - Hardware interrupts
  - Software exceptions / Traps
  - Signals and signal handling
- Describe the essential operation of a modern MMU from a programmer’s standpoint, including:
  - Caching and the TLB
  - Segmentation and paging for virtual memory
- Explain the operation of various memory allocation methods, including:
  - Implicit allocation (garbage collection)
  - Explicit allocation (malloc/free, reference counting, etc.)
- Describe, utilize, and implement a dynamic memory allocation API.
- Describe and utilize the system-level I/O API of a modern operating system, including:
  - File descriptors
  - File I/O
Relationship between Course Outcomes and Program Outcomes

The following Program Outcomes are supported by the above Course Outcomes:

- a. An ability to apply knowledge of computing and mathematics appropriate to the discipline
- b. An ability to analyze a problem, and identify and define the computing requirements appropriate to its solution
- c. An ability to design, implement and evaluate a computer-based system, process, component, or program to meet desired needs
- i. An ability to use current techniques, skills, and tools necessary for computing practices.
- j. An ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices

Prerequisites by Topic

CS 331 Data Structures, CS350 – C/Assembly Programming

Major Topics Covered in the Course

<table>
<thead>
<tr>
<th>Topic</th>
<th>Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Introduction and Syllabus, Course Overview</td>
<td>2</td>
</tr>
<tr>
<td>2. Assembly review / x86 Assembly Primer</td>
<td>2</td>
</tr>
<tr>
<td>3. C: Language basics, Pointers, Arrays, and Structures</td>
<td>8</td>
</tr>
<tr>
<td>4. Processes and the OS, Process management</td>
<td>6</td>
</tr>
<tr>
<td>5. Exceptional Control Flow (signals, signal handling, etc.)</td>
<td>4</td>
</tr>
<tr>
<td>6. Practical: Programming a UNIX shell</td>
<td>2</td>
</tr>
<tr>
<td>7. Caching and Virtual Memory</td>
<td>4</td>
</tr>
<tr>
<td>8. Dynamic Memory Management</td>
<td>6</td>
</tr>
<tr>
<td>9. Practical: Implementing malloc</td>
<td>2</td>
</tr>
<tr>
<td>10. UNIX System Level I/O</td>
<td>4</td>
</tr>
<tr>
<td>11. Interprocess Communication (pipes, message queues, shared memory, etc.)</td>
<td>5</td>
</tr>
<tr>
<td>12. Berkeley sockets API</td>
<td>5</td>
</tr>
<tr>
<td>13. Practical: A Concurrent Server</td>
<td>2</td>
</tr>
<tr>
<td>14. POSIX Threads API</td>
<td>4</td>
</tr>
<tr>
<td>15. Midterm, Recap &amp; Review</td>
<td>4</td>
</tr>
<tr>
<td>Final Exam</td>
<td>-</td>
</tr>
</tbody>
</table>
Assessment Plan for the Course

End of every semester Course Objective Assessments by CS department. End of semester Course Evaluations by IIT. Reviewed every Spring semester by CS Undergraduate Studies Committee for possible updates in the following Fall. Once every 4-5 years a detailed review of all materials for the course is made by the CS Undergraduate Studies Committee.

How Data in the Course is Used to Assess Program Outcomes (unless adequately covered already in the assessment discussion under Criterion 4)

See the assessment discussion under Criterion 4

For a computer science program

Estimate Curriculum Category Content (Semester hours)

<table>
<thead>
<tr>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithms</td>
<td>1</td>
<td></td>
<td>Software design</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Data structures</td>
<td></td>
<td></td>
<td>Concepts of programming languages</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>
COURSE DESCRIPTION

<table>
<thead>
<tr>
<th>Dept., Number</th>
<th>Course Title</th>
<th>Introduction to Algorithms</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS430</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Semester hours 3 Course Coordinator Dr. Sanjiv Kapoor, Professor

Current Catalog Description

Examines the components of sophisticated multi-layer software systems-including device drivers, systems software, applications interfaces, and user interfaces. Explores the design and development of interrupt-driven and event-driven software. Prerequisites: CS 331, CS 350. (2-2-3)

Textbook


References


Course Outcomes

Students should be able to:

- Use big O, omega, and theta notation to give asymptotic upper, lower, and tight bounds on time and space complexity of algorithms.
- Determine the time complexity of simple algorithms, deduce the recurrence relations that describe the time complexity of recursively defined algorithms, and solve simple recurrence relations.
- Design algorithms using the brute-force, greedy, dynamic programming, divide-and-conquer, branch and bound strategies.
- Design algorithms using at least one other algorithmic strategy from the list of topics for this unit.
- Use and implement the fundamental abstract data types -- specifically including hash tables, binary search trees, and graphs -- necessary to solve algorithmic problems efficiently.
- Solve problems using techniques learned in the design of sequential search, binary search, O(N log N) sorting algorithms, and fundamental graph algorithms, including depth-first and breadth-first search, single-source and all-pairs shortest paths, and at least one minimum spanning tree algorithm.
- Demonstrate the following abilities: to evaluate algorithms, to select from a range of possible options, to provide justification for that selection, and to implement the algorithm in simple programming contexts.
- Communicate theoretical and experimental analyses of a set of algorithms (i.e. sorting) in a lab report format.
Relationship between Course Outcomes and Program Outcomes

The following Program Outcomes are supported by the above Course Outcomes:

- a. An ability to apply knowledge of computing and mathematics appropriate to the discipline
- b. An ability to analyze a problem, and identify and define the computing requirements appropriate to its solution
- c. An ability to design, implement and evaluate a computer-based system, process, component, or program to meet desired needs
- d. An ability to communicate effectively with a range of audiences
- e. Recognition of the need for, and an ability to engage in, continuing professional development
- f. An ability to use current techniques, skills, and tools necessary for computing practices.
- g. An ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices
- h. Be prepared to enter a top-ranked graduate program in Computer Science.

Prerequisites by Topic

CS115/CS116 - Object-Oriented Programming: functions, pointers, recursion, classes
CS330 - Discrete Mathematics: sets, functions, counting, proofs
CS331 - Data Structures: abstract data types, lists, stacks, queues, trees

Major Topics Covered in the Course

1. Introduction to Algorithm Design, Complexity analysis including elementary tools 4.5 hours like O-Notations, Recurrence Relations
2. Introduction to Backtracking and Branch and Bound 3 hours
3. Introduction to Dynamic Programming 4.5 hours
4. Divide and Conquer and Greedy Methods (using Traveling Salesman Problem, Knapsack Problem and Optimum Triangulation of Convex Polygons) 3 hours
5. Sorting Methods - Quicksort, Mergesort, Heaps and Heapsort, Lower bound on sorting 4.5 hours
6. Searching I - Hash Functions and Hashing, Union Find 3 hours
7. Searching II-- Binary Search Trees, Balanced Binary Search Trees (AVL Trees, 2-3 trees/ Red-Black trees) 6 hours
8. Graph Algorithms I - Depth First Search, Breadth First search, Bi-connectivity, Topological Sort 4.5 hours
9. Graph Algorithms II - Minimum Spanning Trees, Shortest Paths 4.5 hours
10. String Matching 1.5 hours
11. NP-Complete Problems 3 hours
12. Parallel Model of Computing - Example Sorting (*) 1.5 hours
Midterm Exam 1.5 hours
Final Exam - 45 hours

(*) Optional Topics

Assessment Plan for the Course
End of every semester Course Objective Assessments by CS department. End of semester Course Evaluations by IIT. Reviewed every Spring semester by CS Undergraduate Studies Committee for possible updates in the following Fall. Once every 4-5 years a detailed review of all materials for the course is made by the CS Undergraduate Studies Committee.

How Data in the Course is Used to Assess Program Outcomes (unless adequately covered already in the assessment discussion under Criterion 4)

See the assessment discussion under Criterion 4

For a computer science program
Estimate Curriculum Category Content (Semester hours)

<table>
<thead>
<tr>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithms</td>
<td></td>
<td></td>
<td>Software design</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Data structures</td>
<td>1</td>
<td></td>
<td>Concepts of programming</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>languages</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
COURSE DESCRIPTION

<table>
<thead>
<tr>
<th>Dept., Number</th>
<th>Course Number</th>
<th>Course Title</th>
<th>Semester hours</th>
<th>Course Coordinator</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS440</td>
<td>3</td>
<td>Programming Languages and Translators</td>
<td>3</td>
<td>Dr. Xiang-Yang Li, Associate Professor</td>
</tr>
</tbody>
</table>

Current Catalog Description

Study of commonly used computer programming languages with an emphasis on precision of definition and facility in use. Scanning, parsing, and introduction to compiler design. Use of compiler generating tools. Prerequisite: (CS 330 and CS 351) or CS401 or CS403. (3-0-3) (T)

Textbook
http://dijkstra.cs.iit.edu/cs440-sp08/resources/

References

Course Outcomes

Students should be able to:

- Explain major classes of programming languages: techniques, features, and styles.
  - Know how to use boxed and unboxed variables
  - Be able to use higher order functions.
- How to specify formally the meaning of a language --- to people and to the computer.
  - Use Transition, Typing, and Denotational Semantics to define a language construct.
  - Be able to specify the language of regular expressions.
  - Determine if a grammar is LL, and write a parser for it using recursive descent.
  - Determine if a grammar is LR, and write a parser for it using a parser generator.
  - Describe the algorithm for both LL and LR parser generation.
- Explain Three Powerful Ideas:
  1. Recursion
    - Know how to use both tail recursion and standard recursion.
    - Know how to use higher order functions to eliminate recursion.
  2. Abstraction
    - Know how to create user-defined types.
    - Know how to use functions to model integers.
    - Know how to use trees to model language constructs.
  3. Transformation
    - Know how to interpret a language.
    - Know how to use unification.
- How to choose a language.
How to implement a language.

Emphasis: learn theory and apply it.

Relationship between Course Outcomes and Program Outcomes

The following Program Outcomes are supported by the above Course Outcomes:

- a. An ability to apply knowledge of computing and mathematics appropriate to the discipline
- c. An ability to design, implement and evaluate a computer-based system, process, component, or program to meet desired needs
- h. Recognition of the need for, and an ability to engage in, continuing professional development
- i. An ability to use current techniques, skills, and tools necessary for computing practices.
- j. An ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices
- l. Be prepared to enter a top-ranked graduate program in Computer Science.

Prerequisites by Topic

Experience writing basic programs in more than one computer language and a strong discrete mathematics background.

Major Topics Covered in the Course

1. Course Introduction, Recursion, User Defined Types, Higher Order Functions, Interpreters
   - 7.5 hours
2. Regular Languages, Grammars, LL Parsing, LR Parsing, LR Parsing Tools, Lambda Calculus
   - 9 hours
3. Unification, The Call Stack and the Heap, Transition Semantics, Natural Semantics, Type Semantics
   - 7.5 hours
4. Variables, Parameters, Local State, Objects, Infinite Data, Continuation-Passing Style
   - 9 hours
5. Prolog, Prolog's Cut Operator, Dynamic Prolog, Applications of Prolog
   - 6 hours
6. Meta-Programming
   - 3 hours
Midterm Exams
   - 3 hours
Final Exam
   - 45 hours

Assessment Plan for the Course

End of every semester Course Objective Assessments by CS department. End of semester Course Evaluations by IIT. Reviewed every Spring semester by CS Undergraduate Studies Committee for possible updates in the following Fall. Once every 4-5 years a detailed review of all materials for the course is made by the CS Undergraduate Studies Committee.
How Data in the Course is Used to Assess Program Outcomes (unless adequately covered already in the assessment discussion under Criterion 4)

See the assessment discussion under Criterion 4

For a computer science program
Estimate Curriculum Category Content (Semester hours)

<table>
<thead>
<tr>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithms</td>
<td></td>
<td></td>
<td>Software design</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Data structures</td>
<td></td>
<td></td>
<td>Concepts of programming languages</td>
<td></td>
<td>3</td>
</tr>
</tbody>
</table>
COURSE DESCRIPTION

<table>
<thead>
<tr>
<th>Dept., Number</th>
<th>Course Title</th>
<th>Introduction to Operating Systems</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS450</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Semester hours | 3 |
Course Coordinator | Dr. Xian-He Sun, Professor |

Current Catalog Description
Introduction to operating system concepts—including system organization for uniprocessors and multiprocessors, scheduling algorithms, process management, deadlocks, paging and segmentation, files and protection, and process coordination and communication. Prerequisites: (CS 331 and CS 350) or (CS 331 and ECE 242) or (CS 401 and CS 402) or CS 403. (3-0-3) (T)

Textbook

References

Course Outcomes
Students should be able to:

- Explain the range of requirements that a modern operating system has to address.
- Define the functionality that a modern operating system must deliver to meet a particular need.
- Articulate design tradeoffs inherent in operating system design.
- Explain the concept of a logical layer.
- From the perspective of building operating systems, explain the benefits of building these layers in a hierarchical fashion.
- Describe how the resources of the computer system are managed by software.
- Relate system state to user protection.
- Justify the presence of concurrency within the framework of an operating system.
- Demonstrate the potential run-time problems arising from the concurrent operation of many (possibly a dynamic number of) tasks.
- Summarize the range of mechanisms (at an operating system level) that can be employed to realize concurrent systems and be able to describe the benefits of each.
- Explain the different states that a task may pass through and the data structures needed to support the management of many tasks.
- Compare and contrast the common algorithms used for both preemptive and non-
preemptive scheduling of tasks in operating systems.

- Describe relationships between scheduling algorithms and application domains.
- Investigate the wider applicability of scheduling in such contexts as disk I/O, networking scheduling, and project scheduling.
- Introduce memory hierarchy and cost-performance tradeoffs.
- Explain what virtual memory is and how it is realized in hardware and software.
- Examine the wider applicability and relevance of the concepts of virtual entity and of caching.
- Evaluate the trade-offs in terms of memory size (main memory, cache memory, auxiliary memory) and processor speed.
- Defend the different ways of allocating memory to tasks on the basis of the relative merits of each.
- Summarize the features of an operating system used to provide protection and security, and describe the limitations of each of these.
- Summarize the full range of considerations that support file systems.

Relationship between Course Outcomes and Program Outcomes

The following Program Outcomes are supported by the above Course Outcomes:

- a. An ability to apply knowledge of computing and mathematics appropriate to the discipline
- h. Recognition of the need for, and an ability to engage in, continuing professional development
- j. An ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices
- l. Be prepared to enter a top-ranked graduate program in Computer Science.

Prerequisites by Topic

To be successful in this course you should have substantial programming experience in a high level language (C is ideal) with direct access to the underlying operating system's system call interface. You should be, at minimum, adept at making use of the language's facilities for process control, memory management, I/O, file management, and IPC. Experience with some form of assembly language is also required.

Major Topics Covered in the Course

<table>
<thead>
<tr>
<th>Topic</th>
<th>Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Processes, Threads, and Context Switching</td>
<td>5</td>
</tr>
<tr>
<td>2. System Calls, Interrupts, and Exceptions</td>
<td>5</td>
</tr>
<tr>
<td>3. Kernel and User Modes</td>
<td>5</td>
</tr>
<tr>
<td>4. Scheduling</td>
<td>5</td>
</tr>
<tr>
<td>5. IPC</td>
<td>5</td>
</tr>
<tr>
<td>6. Address spaces, virtual memory and memory management</td>
<td>5</td>
</tr>
</tbody>
</table>
7. I/O and device management  5 hours
8. File systems            5 hours
9. Concurrency            5 hours
Midterm Exam             -
Final Exam               45 hours

Assessment Plan for the Course
End of every semester Course Objective Assessments by CS department. End of semester Course Evaluations by IIT. Reviewed every Spring semester by CS Undergraduate Studies Committee for possible updates in the following Fall. Once every 4-5 years a detailed review of all materials for the course is made by the CS Undergraduate Studies Committee.

How Data in the Course is Used to Assess Program Outcomes (unless adequately covered already in the assessment discussion under Criterion 4)
See the assessment discussion under Criterion 4

For a computer science program
Estimate Curriculum Category Content (Semester hours)

<table>
<thead>
<tr>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithms</td>
<td>1</td>
<td></td>
<td>Software design</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Data structures</td>
<td>1</td>
<td></td>
<td>Concepts of programming languages</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>
COURSE DESCRIPTION

Dept., Number  | CS485 | Course Title | Computers and Society
Semester hours | 3     | Course Coordinator | Charles Bauer, Professor Emeritus

Current Catalog Description

Discussion of the impact of computer technology on present and future society. Historical development of the computer. Social issues raised by cybernetics. Prerequisites: CS 105, COM421 or COM428. (3-0-3) (C)

Textbook

Computers and Society 08/09 Editor: DePalma, Paul McGraw-Hill/Dushkin

References

Course Outcomes

Students should be able to:
- Demonstrate an understanding of the social and professional context in which computing is done.
- Demonstrate an understanding of the basic cultural, social, legal, and ethical issues inherent in the discipline of computing.
- Identify milestones in the development and application of information technology.
- Ask serious questions about the social impact of computing and to evaluate proposed answers to those questions.
- Demonstrate an awareness of the basic legal rights of software and hardware vendors and users, and the ethical values that are the basis for those rights.
- Research the social and ethical issues of a computer related topic from the list in the syllabus.
- Communicate, both orally and in written form, social and ethical issues of a computer related topic from the list in the syllabus.

Relationship between Course Outcomes and Program Outcomes

The following Program Outcomes are supported by the above Course Outcomes:
- d. An ability to function effectively on teams to accomplish a common goal
- e. An understanding of professional, ethical, legal, security, and social issues and responsibilities
- f. An ability to communicate effectively with a range of audiences
- g. An ability to analyze the local and global impact of computing on individuals, organizations and society
Prerequisites by Topic
Broad understanding of information technology and applications.

Major Topics Covered in the Course

1. Topics in *Computers in Society* 11 hours
   - Introduction to Computers in Society and Information Revolutions; The Economy; Work and the Workplace; Computers, People and Social Participation; Social Institutions (property) Law and Politics; Social Values and Risks: Ethics, and Privacy and Preserving the Past; International Perspective and Issues; Philosophical frontiers

2. Social Issues on a broad set of Information Technology related topics such as: 18 hours
   - Computers and Network Security; Computers and Privacy; Computers and Intellectual Property; Computers and Law; Computers and Distance Learning; Computers and Education; Computers and Women and Minorities; etc.

3. History of Computers (by decades) 3 hours

4. Moral and Ethical Issues in Computers and Society and the ACM code of Ethics 9 hours

Orientation, Student Portfolio (for assessment purposes), Midterm 4 hours
Final Exam - 45 hours

Assessment Plan for the Course
End of every semester Course Objective Assessments by CS department. End of semester Course Evaluations by IIT. Reviewed every Spring semester by CS Undergraduate Studies Committee for possible updates in the following Fall. Once every 4-5 years a detailed review of all materials for the course is made by the CS Undergraduate Studies Committee.

How Data in the Course is Used to Assess Program Outcomes (unless adequately covered already in the assessment discussion under Criterion 4)
See the assessment discussion under Criterion 4

*For a computer science program*

Estimate Curriculum Category Content (Semester hours)

<table>
<thead>
<tr>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithms</td>
<td></td>
<td></td>
<td>Software design</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Data structures</td>
<td></td>
<td></td>
<td>Concepts of programming languages</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
COURSE DESCRIPTION

<table>
<thead>
<tr>
<th>Dept., Number</th>
<th>CS487</th>
<th>Course Title</th>
<th>Software Engineering</th>
</tr>
</thead>
<tbody>
<tr>
<td>Semester hours</td>
<td>3</td>
<td>Course Coordinator</td>
<td>Dr. Bogdan Korel, Associate Professor</td>
</tr>
</tbody>
</table>

Current Catalog Description

Study of the principles and practices of software engineering. Topics include software quality concepts, process models, software requirements analysis, design methodologies, software testing, and software maintenance. Hands-on experience building a software system using the waterfall life cycle model. Students working in teams develop all life cycle deliverables: requirements document, specification and design documents, system code, test plan, and user manuals. Prerequisite: CS 331 or CS 401 or CS 403. (3-0-3) (T) (C)

Textbook


References

Course Outcomes

Students should be able to:
- Understand and explain software development as a series of engineering activities, and processes.
- Demonstrate software development team-working skills.
- Analyze client/user needs.
- Select an appropriate life cycle and process model for development of a software product.
- Explain the importance of software quality evaluation activities.
- Develop a series of software life-cycle deliverables.
- Develop representations/models and descriptions of an evolving software product for inclusion in a requirements specification document.
- Build a multi-level design model and evaluate software design alternatives
- Design, execute, and log multi-level software tests.
- Describe the role that tools can play in the software life cycle.
- Communicate, verbally and in writing, the deliverables of a software development project.

Relationship between Course Outcomes and Program Outcomes

The following Program Outcomes are supported by the above Course Outcomes:
- b. An ability to analyze a problem, and identify and define the computing requirements
appropriate to its solution
• c. An ability to design, implement and evaluate a computer-based system, process, component, or program to meet desired needs
• d. An ability to function effectively on teams to accomplish a common goal
• e. An understanding of professional, ethical, legal, security, and social issues and responsibilities
• f. An ability to communicate effectively with a range of audiences
• h. Recognition of the need for, and an ability to engage in, continuing professional development
• i. An ability to use current techniques, skills, and tools necessary for computing practices.
• k. An ability to apply design and development principles in the construction of software systems of varying complexity
• l. Be prepared to enter a top-ranked graduate program in Computer Science.

Prerequisites by Topic

Experience in developing basic programs in any computer language
Have an understanding of, and be able to apply, the essential data structures and algorithms used in computer science.

Major Topics Covered in the Course

<table>
<thead>
<tr>
<th>Topic</th>
<th>Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. The problem statement, developer-client interactions. Overview of software engineering - life cycle models, software deliverables.</td>
<td>3 hours</td>
</tr>
<tr>
<td>2. Software development team concepts, team organization, team structures. Project management, the project plan.</td>
<td>3 hours</td>
</tr>
<tr>
<td>3. Requirements analysis, methods, models. For example, structured analysis with use of data flow diagrams, data dictionary, and entity-relationship diagrams.</td>
<td>7 hours</td>
</tr>
<tr>
<td>4. Software specification, methods, and models. For example, structured analysis with use of process specifications, state transition diagrams.</td>
<td>3.5 hours</td>
</tr>
<tr>
<td>5. Preliminary design concepts, methods, and models. For example, structured analysis with use of structure charts, procedural abstractions. Concepts of top down decomposition, bottom-up composition, abstraction, coupling, cohesion, modularity, information hiding, reuse, architectural styles.</td>
<td>6.5 hours</td>
</tr>
<tr>
<td>6. Detailed design concepts, methods and models. For example, structured analysis with use of PDL (Program Design Language. Algorithm, and data structure design.</td>
<td>2.5 hours</td>
</tr>
<tr>
<td>7. Object concepts. Object-oriented analysis, nature of the approach, models. For example, Coad/Yourdon analysis model with use of class diagrams, class hierarchies, attribute, and service specifications. Role of use cases. Use of modeling languages such as UML. Object-oriented design approaches, for example Coad/Yourdon's 4-layer object-oriented design model.</td>
<td>4.5 hours</td>
</tr>
<tr>
<td>8. Software implementation, transition from design to code.</td>
<td>1 hour</td>
</tr>
<tr>
<td>9. Software testing and evaluation. Black and white box test design strategies and</td>
<td>6.5 hours</td>
</tr>
</tbody>
</table>
related techniques, testing at multiple levels, regression test.

10. Software quality, reviews, and metrics. 3 hours

11. Software maintenance and re-engineering. Types of maintenance, role of configuration management, legacy code, tool support for maintenance. 1.5 hours

12. Selected Topics 1.5 hours
Midterm Exam 1.5 hours
Final Exam -

**Assessment Plan for the Course**

End of every semester Course Objective Assessments by CS department. End of semester Course Evaluations by IIT. Reviewed every Spring semester by CS Undergraduate Studies Committee for possible updates in the following Fall. Once every 4-5 years a detailed review of all materials for the course is made by the CS Undergraduate Studies Committee.

How Data in the Course is Used to Assess Program Outcomes (unless adequately covered already in the assessment discussion under Criterion 4)

See the assessment discussion under Criterion 4

**For a computer science program**

<table>
<thead>
<tr>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithms</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Data structures</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
</tr>
</thead>
<tbody>
<tr>
<td>Software design</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Concepts of programming languages</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

45 hours
Elective Computer Science Courses
COURSE DESCRIPTION

<table>
<thead>
<tr>
<th>Dept., Number</th>
<th>Course Title</th>
<th>Computer Graphics</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS411</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Semester hours | 3 | Course Coordinator | Dr. Gady Agam, Associate Professor |

Current Catalog Description


Prerequisites: CS 331 or CS401 or CS403. (3-0-3) (T)

Textbook


References


Course Outcomes

The following are the main objectives of the course:

- Provide overview of computer graphics.
- Provide understanding of basic concepts, mathematical models, techniques, and algorithms used in computer graphics in two and three dimensions.
- Provide graphics programming experience with OpenGL.

Students should be able to:

- Describe and understand the main areas of computer graphics, graphics software, and graphics hardware.
- Demonstrate an understanding of the basic concepts, mathematical models, techniques and algorithms relating to raster graphics. The students should be able to implement basic algorithms and modify them if necessary.
- Demonstrate an understanding of the basic concepts, syntax, and techniques behind the OpenGL graphics library. The students should be able to write graphics programs by using this software library.
- Demonstrate an understanding of the basic concepts, mathematical models, techniques and algorithms relating to 2D and 3D modeling and viewing. The students should be able to implement basic algorithms and modify them if necessary. They should be able to use OpenGL in this context.
- Demonstrate an understanding of the basic concepts, mathematical models, techniques and algorithms relating to 3D object representation. The students should be able to implement basic algorithms and modify them if necessary.
• Demonstrate an understanding of the basic concepts, mathematical models, techniques and algorithms relating to Color. The students should be able to implement basic algorithms and modify them if necessary.

• Demonstrate an understanding of the basic concepts, mathematical models, techniques and algorithms relating to Illumination models and surface rendering. The students should be able to implement basic algorithms and modify them if necessary. They should be able to use OpenGL in this context.

Relationship between Course Outcomes and Program Outcomes

The following Program Outcomes are supported by the above Course Outcomes:

• a. An ability to apply knowledge of computing and mathematics appropriate to the discipline
• c. An ability to design, implement and evaluate a computer-based system, process, component, or program to meet desired needs
• i. An ability to use current techniques, skills, and tools necessary for computing practices.
• j. An ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices

Prerequisites by Topic

Math - Calculus, Linear algebra
Programming - Data structures and algorithms, C/C++

Major Topics Covered in the Course

<table>
<thead>
<tr>
<th>Major Topics</th>
<th>Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Introduction: overview of computer graphics, overview of graphics hardware and software</td>
<td>1.5 hours</td>
</tr>
<tr>
<td>2. Introduction to graphics programming with OpenGL: overview, concepts, syntax, libraries, basic drawing, state management</td>
<td>3 hours</td>
</tr>
<tr>
<td>3. Raster graphics: line and conic sections drawings, area filling, character generation, image operations, object attributes, antialiasing.</td>
<td>4.5 hours</td>
</tr>
<tr>
<td>4. 2D modeling and viewing: geometric transformations, homogeneous coordinates, affine transformation, line and polygon display</td>
<td>6 hours</td>
</tr>
<tr>
<td>5. Introduction to 3D Rendering with OpenGL: 3d rendering concepts, 3d modeling and viewing in OpenGL</td>
<td>6 hours</td>
</tr>
<tr>
<td>6. 3D modeling and viewing: 3D transformations, the 3D viewing pipeline, projections, clipping, visible surface detection, hierarchical modeling</td>
<td>7.5 hours</td>
</tr>
<tr>
<td>7. 3D object representation: polygonal surfaces, quadric surfaces, cubic splines, Bezier curves and surfaces, B-spline curves and surfaces, NURBS, CSG, octrees, BSP trees, other representations</td>
<td>7.5 hours</td>
</tr>
<tr>
<td>8. Color, illumination models, and surface rendering: basic illumination models, polygon rendering, ray tracing, texture and bump mapping, displaying light intensities, dithering, color models, LUTs, blending</td>
<td>7.5 hours</td>
</tr>
<tr>
<td>9. Midterm, Recap &amp; Review</td>
<td>1.5 hours</td>
</tr>
<tr>
<td>Final Exam</td>
<td>-</td>
</tr>
</tbody>
</table>
Assessment Plan for the Course

End of every semester Course Objective Assessments by CS department. End of semester Course Evaluations by IIT. Reviewed every Spring semester by CS Undergraduate Studies Committee for possible updates in the following Fall. Once every 4-5 years a detailed review of all materials for the course is made by the CS Undergraduate Studies Committee.

How Data in the Course is Used to Assess Program Outcomes (unless adequately covered already in the assessment discussion under Criterion 4)

See the assessment discussion under Criterion 4

For a computer science program

Estimate Curriculum Category Content (Semester hours)

<table>
<thead>
<tr>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithms</td>
<td></td>
<td>1.5</td>
<td>Software design</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Data structures</td>
<td></td>
<td>1.5</td>
<td>Concepts of programming languages</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
COURSE DESCRIPTION

<table>
<thead>
<tr>
<th>Dept., Number</th>
<th>CS422</th>
<th>Course Title</th>
<th>Introduction to Data Mining</th>
</tr>
</thead>
<tbody>
<tr>
<td>Semester hours</td>
<td>3</td>
<td>Course Coordinator</td>
<td>Dr. Nazli Goharian, Clinical Associate Professor</td>
</tr>
</tbody>
</table>

Current Catalog Description

This course will provide an introductory look at concepts and techniques in the field of data mining. After covering the introduction and terminologies to Data Mining, the techniques used to explore the large quantities of data for the discovery of meaningful rules and knowledge such as market basket analysis, nearest neighbor, decision trees, neural networks, and clustering are covered. The students learn the material by implementing different techniques throughout the semester (3-0-3).

Textbook

J. Han, M. Kamber. Data Mining Concepts and Techniques, Morgan Kaufmann, 2006

References

Course Outcomes

Students should be able to:

- Explain the Data Mining motivation and applications.
- Explain the Data Mining Architecture.
- Explain Data Preprocessing motivation and techniques.
- Explain various Data Mining algorithms such as Naïve Bayes, Neural Networks, Decision Tree, Association-Rules, and Clustering.
- Explain the scalability issues for each of the algorithms discussed in the class and how they can be modified for scalability.
- Design and implement data mining systems using various data pre-processing techniques and mining algorithms.
- Apply the research ideas into their experiments in building data mining systems.

Relationship between Course Outcomes and Program Outcomes

The following Program Outcomes are supported by the above Course Outcomes:

- a. An ability to apply knowledge of computing and mathematics appropriate to the discipline
- c. An ability to design, implement and evaluate a computer-based system, process, component, or program to meet desired needs
- d. An ability to function effectively on teams to accomplish a common goal
- f. An ability to communicate effectively with a range of audiences
- j. An ability to apply mathematical foundations, algorithmic principles, and computer
Prerequisites by Topic
Data Structures, Algorithm and Strong Object Oriented Programming.

Major Topics Covered in the Course

<table>
<thead>
<tr>
<th>Topic</th>
<th>Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Introduction to Data Mining</td>
<td>1.5</td>
</tr>
<tr>
<td>2. Data preprocessing</td>
<td>3</td>
</tr>
<tr>
<td>3. Classification &amp; Cross Validation</td>
<td>3</td>
</tr>
<tr>
<td>4. Evaluation</td>
<td>3</td>
</tr>
<tr>
<td>5. Naive Bayes</td>
<td>3</td>
</tr>
<tr>
<td>6. Neural Networks</td>
<td>4.5</td>
</tr>
<tr>
<td>7. Decision Tree</td>
<td>4.5</td>
</tr>
<tr>
<td>8. Rule Based Classification</td>
<td>3</td>
</tr>
<tr>
<td>10. Ensemble Methods</td>
<td>3</td>
</tr>
<tr>
<td>11. Association rules</td>
<td>3</td>
</tr>
<tr>
<td>12. Cluster analysis</td>
<td>4.5</td>
</tr>
<tr>
<td>13. Students Presentations</td>
<td>6</td>
</tr>
<tr>
<td>Final Exam</td>
<td></td>
</tr>
<tr>
<td></td>
<td>45</td>
</tr>
</tbody>
</table>

Assessment Plan for the Course

End of every semester Course Objective Assessments by CS department. End of semester Course Evaluations by IIT. Reviewed every Spring semester by CS Undergraduate Studies Committee for possible updates in the following Fall. Once every 4-5 years a detailed review of all materials for the course is made by the CS Undergraduate Studies Committee.

How Data in the Course is Used to Assess Program Outcomes (unless adequately covered already in the assessment discussion under Criterion 4)
See the assessment discussion under Criterion 4

For a computer science program

Estimate Curriculum Category Content (Semester hours)

<table>
<thead>
<tr>
<th>Category</th>
<th>Core</th>
<th>Advanced</th>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithms</td>
<td>1</td>
<td></td>
<td>Software design</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Data structures</td>
<td>1</td>
<td></td>
<td>Concepts of programming languages</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
COURSE DESCRIPTION

<table>
<thead>
<tr>
<th>Dept., Number</th>
<th>CS425</th>
<th>Course Title</th>
<th>Database Organization</th>
</tr>
</thead>
<tbody>
<tr>
<td>Semester hours</td>
<td>3</td>
<td>Course Coordinator</td>
<td>Dr. Nazli Goharian, Clinical Associate Professor</td>
</tr>
</tbody>
</table>

Current Catalog Description

Overview of database architectures, including the Relational, Hierarchical, Network, and Object Models. Database interfaces, including the SQL query language. Database design using the Entity-Relationship Model. Issues such as security, integrity, and query optimization. Prerequisite: CS 331 or CS 401 or CS 403. (3-0-3) (T) (C)

Textbook


or


References

Course Outcomes

Students should be able to:

- Design and model a design scenario using relational data modeling, which includes:
  - Analyze the design anomalies.
  - Construct Entity Relationship Diagram.
  - Analyze and Construct Functional Dependencies for the business rules.
  - Analyze Functional Dependencies to identify Primary keys.
  - Analyze and Perform Normalization and Normal Forms.
  - Define referential integrities.
  - Create relational database design schemas in 3-NF/BCNF for a design scenario of the size of ca. 8-10 tables.

- Solve abstract relational language, such as relational algebra problems.

- Solve database transactions by using Structured Query Language (SQL), used by RDBMS.

- Explain the general concept of the additional topics such as: Query Optimizations, Concurrency Control, Recovery, structured data and text, and data warehousing.

- Implement a relational database application, using a commercial/ open source RDBMS (Such as Oracle or mysql). This includes both the design and the implementation of an application that uses a relational database management system for the storage of the data.
and provides a user interface for the insertion, deletion, update and query of the data in this database by a user.

Relationship between Course Outcomes and Program Outcomes

The following Program Outcomes are supported by the above Course Outcomes:

- a. An ability to apply knowledge of computing and mathematics appropriate to the discipline
- c. An ability to design, implement and evaluate a computer-based system, process, component, or program to meet desired needs
- d. An ability to function effectively on teams to accomplish a common goal
- f. An ability to communicate effectively with a range of audiences
- i. An ability to use current techniques, skills, and tools necessary for computing practices.
- j. An ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices
- k. An ability to apply design and development principles in the construction of software systems of varying complexity

Prerequisites by Topic

Data Structures, Algorithm and Strong Object Oriented Programming.

Major Topics Covered in the Course

1. Introduction 1.5 hours
2. Relational Model 4.5 hours
3. Relational Algebra 4.5 hours
4. SQL 4.5 hours
5. Database Design 4.5 hours
6. Query Optimization 4.5 hours
7. Recovery and Concurrency Control 4.5 hours
8. Integration of Structured Data and Text 4.5 hours
9. Special Topics: Data Warehousing, Data Mining 4.5 hours
10. Midterm and review 3 hours
   Final Exam - 45 hours

Assessment Plan for the Course

End of every semester Course Objective Assessments by CS department. End of semester Course Evaluations by IIT. Reviewed every Spring semester by CS Undergraduate Studies Committee for possible updates in the following Fall. Once every 4-5 years a detailed review of all materials for the course is made by the CS Undergraduate Studies Committee.

How Data in the Course is Used to Assess Program Outcomes (unless adequately covered already in the assessment discussion under Criterion 4)
See the assessment discussion under Criterion 4

*For a computer science program*

Estimate Curriculum Category Content (Semester hours)

<table>
<thead>
<tr>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithms</td>
<td>1</td>
<td></td>
<td>Software design</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Data structures</td>
<td>1</td>
<td></td>
<td>Concepts of programming languages</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
COURSE DESCRIPTION

<table>
<thead>
<tr>
<th>Dept., Number</th>
<th>Course Title</th>
<th>Introduction to Information Retrieval Systems</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS429</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Semester hours</td>
<td>3</td>
<td>Course Coordinator: Dr. Nazli Goharian, Clinical Associate Professor</td>
</tr>
</tbody>
</table>

Current Catalog Description

Overview of fundamental issues of information retrieval with theoretical foundations. The Information-retrieval techniques and theory, covering both effectiveness and run-time performance of information-retrieval systems are covered. The focus is on algorithms and heuristics used to find documents relevant to the user request and to find them fast. The course covers the architecture and components of the search engine such as parser, stemmer, index builder, and query processor. The students learn the material by building a prototype of such a search engine. Prerequisite: CS331 or CS401 and strong programming knowledge. (3-0-3) (T) (C)

Textbook


References

Course Outcomes

Students should be able to:
- Explain the information retrieval storage methods (Inverted Index and Signature Files)
- Explain retrieval models, such as Boolean model, Vector Space model, Probabilistic model, Inference Networks, and Neural Networks.
- Explain retrieval utilities such as Stemming, Relevance Feedback, N-gram, Clustering, and Thesauri, and Parsing and Token recognition.
- Design and implement a search engine prototype using the storage methods, retrieval models and utilities.
- Apply the research ideas into their experiments in building a search engine prototype

Relationship between Course Outcomes and Program Outcomes

The following Program Outcomes are supported by the above Course Outcomes:
- a. An ability to apply knowledge of computing and mathematics appropriate to the discipline
- c. An ability to design, implement and evaluate a computer-based system, process, component, or program to meet desired needs
- d. An ability to function effectively on teams to accomplish a common goal
- f. An ability to communicate effectively with a range of audiences
- i. An ability to use current techniques, skills, and tools necessary for computing practices.
- j. An ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices
- k. An ability to apply design and development principles in the construction of software systems of varying complexity

Prerequisites by Topic
Data Structures, Algorithm and Strong Object Oriented Programming.

Major Topics Covered in the Course

<table>
<thead>
<tr>
<th>Topic</th>
<th>Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Introduction, Overview of IR</td>
<td>1.5 hours</td>
</tr>
<tr>
<td>2. IR Utilities: Parser/Tokenizer, phrase Recognition, Stemming, N-Grams</td>
<td>3 hours</td>
</tr>
<tr>
<td>3. Efficiency: Indexing - inverted index, memory based and sort inversion; Signature Files</td>
<td>3 hours</td>
</tr>
<tr>
<td>4. IR Strategies and Models: Boolean, Vector Space Model; Similarity Measures in Information Retrieval, Pivoted Normalizations</td>
<td>3 hours</td>
</tr>
<tr>
<td>5. IR Evaluation</td>
<td>3 hours</td>
</tr>
<tr>
<td>6. IR Strategy: Probabilistic Model</td>
<td>3 hours</td>
</tr>
<tr>
<td>7. IR Utility: Relevance Feedback and other Query Expansions</td>
<td>3 hours</td>
</tr>
<tr>
<td>8. Efficiency : Compression</td>
<td>3 hours</td>
</tr>
<tr>
<td>9. Efficiency: Top Docs, Query Threshold</td>
<td>3 hours</td>
</tr>
<tr>
<td>10. Clustering</td>
<td>3 hours</td>
</tr>
<tr>
<td>11. IR Strategy: Language Models</td>
<td>3 hours</td>
</tr>
<tr>
<td>12. World Wide Web</td>
<td>3 hours</td>
</tr>
<tr>
<td>13. IR Utility: Passage Based Retrieval</td>
<td>3 hours</td>
</tr>
<tr>
<td>14. Efficiency: Duplicate Document Detection</td>
<td>3 hours</td>
</tr>
<tr>
<td>15. Relational Approach</td>
<td>1.5 hours</td>
</tr>
<tr>
<td>16. Student Presentations</td>
<td>3 hours</td>
</tr>
<tr>
<td>Final Exam</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>45 hours</td>
</tr>
</tbody>
</table>

Assessment Plan for the Course

End of every semester Course Objective Assessments by CS department. End of semester Course Evaluations by IIT. Reviewed every Spring semester by CS Undergraduate Studies Committee for possible updates in the following Fall. Once every 4-5 years a detailed review of all materials for the course is made by the CS Undergraduate Studies Committee.

How Data in the Course is Used to Assess Program Outcomes (unless adequately covered already in the assessment discussion under Criterion 4)

See the assessment discussion under Criterion 4
For a computer science program
Estimate Curriculum Category Content (Semester hours)

<table>
<thead>
<tr>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithms</td>
<td>1</td>
<td></td>
<td>Software design</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>Data structures</td>
<td></td>
<td>1</td>
<td>Concepts of programming languages</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
COURSE DESCRIPTION

<table>
<thead>
<tr>
<th>Dept., Number</th>
<th>CS441</th>
<th>Course Title</th>
<th>Current Topics in Programming Languages</th>
</tr>
</thead>
<tbody>
<tr>
<td>Semester hours</td>
<td>3</td>
<td>Course Coordinator</td>
<td>Dr. Tzilla Elrad, Research Professor</td>
</tr>
</tbody>
</table>

Current Catalog Description

New topics in programming language design such as concepts of concurrent and distributed programming, communicating sequential processes, and functional programming. System development tools and language features for programming. An introduction to programming language semantics. Prerequisite: CS 331 or CS 401 or CS 403. (3-0-3) (T)

Textbook


References

See http://www.cs.iit.edu/~cs441/index.html

Course Outcomes

The course is basically language independent. Any language that can support the course goals may be selected. An example of a choice language might be Java. A student should be able to:

- Outline the evolution of the architectural neutral, secure, OO programming languages in order to illustrate how this evolution has led to the occurrence of the JAVA programming model. The course builds on the students' knowledge of Object Oriented Programming concepts, which is a prerequisite for the course.
- Design, implement, test, and debug Applets, Servlets, and Applications.
- Design and implement Graphical User Interfaces.
- Learn the programming language mechanisms that support distribution transparency and development of distributed applications.
- Recognize the underlying concurrency language model; Multithreading and monitor-based concurrency model.
- Demonstrate the supportive language constructs and mechanisms for the design and development of 3-tier architectures; server-side programming.

Relationship between Course Outcomes and Program Outcomes

The following Program Outcomes are supported by the above Course Outcomes:

- b. An ability to analyze a problem, and identify and define the computing requirements appropriate to its solution
- c. An ability to design, implement and evaluate a computer-based system, process, component, or program to meet desired needs
- h. Recognition of the need for, and an ability to engage in, continuing professional development
- i. An ability to use current techniques, skills, and tools necessary for computing practices.
- k. An ability to apply design and development principles in the construction of software systems of varying complexity.

Prerequisites by Topic

Strong object-oriented programming experience.

Major Topics Covered in the Course

| 1. Object-Oriented Programming Overview | 3 hours |
| 2. Event-driven programming for building GUI | 3 hours |
| 3. Security and Web Servers | 3 hours |
| 4. Multithreading | 3 hours |
| 5. Animation and Serialization | 3 hours |
| 6. Database Connectivity | 3 hours |
| 5. Networking and Multicasting | 6 hours |
| 6. Client/Server Models | 18 hours |
| 7. Aspect-Oriented Programming | 3 hours |
| Total hours | 45 hours |

Assessment Plan for the Course

End of every semester Course Objective Assessments by CS department. End of semester Course Evaluations by IIT. Reviewed every Spring semester by CS Undergraduate Studies Committee for possible updates in the following Fall. Once every 4-5 years a detailed review of all materials for the course is made by the CS Undergraduate Studies Committee.

How Data in the Course is Used to Assess Program Outcomes (unless adequately covered already in the assessment discussion under Criterion 4)

See the assessment discussion under Criterion 4

For a computer science program

Estimate Curriculum Category Content (Semester hours)

<table>
<thead>
<tr>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithms</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Data structures</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Software design</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Concepts of programming languages</td>
<td></td>
<td>3</td>
</tr>
</tbody>
</table>
COURSE DESCRIPTION

<table>
<thead>
<tr>
<th>Dept., Number</th>
<th>Course Title</th>
<th>Object Oriented Design and Programming</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS445</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Semester hours</th>
<th>Course Coordinator</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>Dr. Bogdan Korel, Associate Professor</td>
</tr>
</tbody>
</table>

Current Catalog Description

Introduction to methodologies for object-oriented design and programming. Examines the object model and how it is realized in various object-oriented languages. Focuses on methods for developing and implementing object-oriented systems. Prerequisite: CS 331 or CS 401 or CS 403 (3-0-3) (T)

Textbook


References

See http://www.cs.iit.edu/~cs445

Course Outcomes

Students should be able to:

- Explain and justify the principles of Object Oriented concepts (review abstraction & abstract data types, encapsulation, inheritance, polymorphism, aggregation)
- Analyze and identify the strengths (and weaknesses) of in-depth areas of the Object Oriented paradigm.
- Analyze, explain, & compare the qualities of Object Oriented languages and how well they support the object model.
- Explain and analyze the key points of Object Oriented analysis.
- Explain and analyze the key points of Object Oriented design.
- Design, implement, test and debug multi-phased Object Oriented application.
- Explain and utilize contemporary Object Oriented methodologies (data-driven methodology and behavior-driven methodology)
- Utilize contemporary notation (Unified Modeling Language) to express the artifacts of Object Oriented Analysis & Design (class design, class relationships, object interaction, object states, etc.)
- Perform Object Oriented Analysis & Design on a real-world problem.
- Explain and Utilize Complex Design Patterns.
- Create an implementation of the resultant Object Oriented design.
- Examine new & contemporary concepts in Object Orientation.
• Communicate the deliverables of a software development project.

Relationship between Course Outcomes and Program Outcomes

The following Program Outcomes are supported by the above Course Outcomes:

- b. An ability to analyze a problem, and identify and define the computing requirements appropriate to its solution
- c. An ability to design, implement and evaluate a computer-based system, process, component, or program to meet desired needs
- f. An ability to communicate effectively with a range of audiences.
- i. An ability to use current techniques, skills, and tools necessary for computing practices.
- j. An ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices
- k. An ability to apply design and development principles in the construction of software systems of varying complexity.

Prerequisites by Topic

Strong object-oriented programming experience.

Major Topics Covered in the Course

<table>
<thead>
<tr>
<th></th>
<th>Review of The Terminology And Fundamentals Of Object Oriented Concepts</th>
<th>1.0 hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>Abstractions/Abstract Data Types/Encapsulation/Information</td>
<td>3.5 hours</td>
</tr>
<tr>
<td></td>
<td>Hiding/Coupling/Cohesion</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Object Oriented Hierarchies - Advances Topics on</td>
<td>3.5 hours</td>
</tr>
<tr>
<td></td>
<td>Inheritance/Polymorphism/Dynamic Binding/Aggregations</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>&quot;Interface&quot; Class Concepts</td>
<td>2 hours</td>
</tr>
<tr>
<td>5</td>
<td>Object Oriented Languages – Survey, Features</td>
<td>2.5 hours</td>
</tr>
<tr>
<td>6</td>
<td>Characteristics of Objects (Object Relationships, Object Interactions, Instantiation, etc.)</td>
<td>2.5 hours</td>
</tr>
<tr>
<td>7</td>
<td>Object Oriented Analysis &amp; Design - Concepts, Methodologies, Unified Modeling Language</td>
<td>6 hours</td>
</tr>
<tr>
<td>8</td>
<td>Structural Modeling (Class Diagram)</td>
<td>3 hours</td>
</tr>
<tr>
<td>9</td>
<td>Behavioral Modeling (Interaction Diagram, State Diagram)</td>
<td>2 hours</td>
</tr>
<tr>
<td>10</td>
<td>Object-Oriented Design Patterns - Understanding &amp; Usage</td>
<td>3.5 hours</td>
</tr>
<tr>
<td>11</td>
<td>End-To-End Case Study of Object-Oriented Analysis &amp; Design</td>
<td>3 hours</td>
</tr>
<tr>
<td>12</td>
<td>Object Oriented Detailed Design</td>
<td>2 hours</td>
</tr>
<tr>
<td>13</td>
<td>Object Oriented Analysis &amp; Design in Large Scale Projects</td>
<td>2 hours</td>
</tr>
<tr>
<td>14</td>
<td>Use Of Persistence &amp; Databases In an Object Oriented Application</td>
<td>2 hours</td>
</tr>
<tr>
<td>15</td>
<td>Contemporary Object Oriented Topics, Including Multi-Threaded Objects</td>
<td>4 hours</td>
</tr>
<tr>
<td>16</td>
<td>Course Administration &amp; Mid-Term Exam</td>
<td>2.5 hours</td>
</tr>
<tr>
<td>17</td>
<td>Final Exam</td>
<td>45 hours</td>
</tr>
</tbody>
</table>

Assessment Plan for the Course
End of every semester Course Objective Assessments by CS department. End of semester Course Evaluations by IIT. Reviewed every Spring semester by CS Undergraduate Studies Committee for possible updates in the following Fall. Once every 4-5 years a detailed review of all materials for the course is made by the CS Undergraduate Studies Committee.

How Data in the Course is Used to Assess Program Outcomes (unless adequately covered already in the assessment discussion under Criterion 4)

See the assessment discussion under Criterion 4

For a computer science program
Estimate Curriculum Category Content (Semester hours)

<table>
<thead>
<tr>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithms</td>
<td></td>
<td></td>
<td>Software design</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Data structures</td>
<td></td>
<td></td>
<td>Concepts of programming languages</td>
<td></td>
<td>3</td>
</tr>
</tbody>
</table>
COURSE DESCRIPTION

<table>
<thead>
<tr>
<th>Dept., Number</th>
<th>Course Title</th>
<th>Distributed Objects</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS447</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Semester hours</th>
<th>Course Coordinator</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>Dr. Shangping Ren, Assistant Professor</td>
</tr>
</tbody>
</table>

Current Catalog Description

This course provides an introduction to the architecture, analysis, design, and implementation of distributed, multi-tier applications using distributed object technology. The course focuses on the services and facilities provided by an Object Request Broker (ORB). Students will use a commercially available ORB and Database Management System to develop distributed object applications. Prerequisite: CS 445. (3-0-3) (T) (C)

Textbook


References

See http://www.cs.iit.edu/~cs447

Course Outcomes

Students should be able to:
- Understand the basic concept of distributed systems and distributed objects
- Understand the principles of Object-Oriented Middleware and common design problems for distributed systems
- Understand advantages and disadvantages of various multi-tier software architectures
- Use IDL to define application interfaces
- Use business objects to construct software applications
- Understand functions of an Object Request Broker (ORB), common distributed services, common distributed messaging styles, multiple mechanisms for providing object persistence used in distributed applications
- Understand and be able to use iterative, use case driven methodology in component-based software development
- Implement a distributed, multi-tier application using distributed object technology
- Acquire software development team-working skills using a use case driven, architecture-centric, iterative software development process

Relationship between Course Outcomes and Program Outcomes
The following Program Outcomes are supported by the above Course Outcomes:

- b. An ability to analyze a problem, and identify and define the computing requirements appropriate to its solution
- c. An ability to design, implement and evaluate a computer-based system, process, component, or program to meet desired needs
- h. Recognition of the need for, and an ability to engage in, continuing professional development
- i. An ability to use current techniques, skills, and tools necessary for computing practices.
- k. An ability to apply design and development principles in the construction of software systems of varying complexity.

Prerequisites by Topic

- Fundamental aspects of the object-oriented model: abstraction, encapsulation, inheritance, and aggregation.
- Fundamental aspects of developing object-oriented software: requirements, analysis, design, implementation, testing, and deployment.
- Experience writing object-oriented software using a common object-oriented programming language.
- Experience using a relational database management system.

Major Topics Covered in the Course

| 1. Course Introduction       | 1.0 hour |
| 2. Software Architectures, and Business Object Architecture | 4.0 hours |
| 3. OMG Object Management Architecture, and CORBA Overview | 1.5 hours |
| 4. Interface Definition Languages, and Distributed Programming | 4.5 hours |
| 5. Project Overview | 1.0 hour |
| 6. Business Object, and Use Case Modeling | 3.0 hours |
| 7. Common Distributed Services | 1.5 hours |
| 8. Directory Services | 1.5 hours |
| 9. Persistence | 3.0 hours |
| 10. Midterm Exam | 3.0 hours |
| 11. Object to Relational Mapping, and Persistence Frameworks | 3.0 hours |
| 12. Event, Notification, and Messaging Services | 3.0 hours |
| 13. Object Database Management Systems | 3.0 hours |
| 14. Transaction Service | 3.0 hours |
| 15. Object Activation | 3.0 hours |
| 16. Application Servers, and Component Frameworks | 3.0 hours |
| 17. Future Trends | 3.0 hours |
| Total | 45 hours |

Assessment Plan for the Course

End of every semester Course Objective Assessments by CS department. End of semester Course Evaluations by IIT. Reviewed every Spring semester by CS Undergraduate Studies Committee for possible updates in the following Fall. Once every 4-5 years a detailed review of all materials for the course is made by the CS Undergraduate Studies Committee.
How Data in the Course is Used to Assess Program Outcomes (unless adequately covered already in the assessment discussion under Criterion 4)

See the assessment discussion under Criterion 4

*For a computer science program*

Estimate Curriculum Category Content (Semester hours)

<table>
<thead>
<tr>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithms</td>
<td></td>
<td></td>
<td>Software design</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Data structures</td>
<td></td>
<td></td>
<td>Concepts of programming languages</td>
<td>3</td>
<td></td>
</tr>
</tbody>
</table>
COURSE DESCRIPTION

<table>
<thead>
<tr>
<th>Dept., Number</th>
<th>Course Title</th>
<th>Data Communications</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS455</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Semester hours: 3  
Course Coordinator: Dr. Peng-Jun Wan, Associate Professor

Current Catalog Description

Introduction to data communication concepts and facilities with an emphasis on protocols and interface specifications. Focuses on the lower four layers of the ISO-OSI reference model.  
Prerequisite: CS 450. (3-0-3) (T)

Textbook


References

Course Outcomes

Students should be able to:

- Understand the operation of multi-layered protocols, particularly the OSI and Internet models/architectures, and how standards evolve.
- Describe the difference between different network topologies, including packet and circuit switched, LANs and WANs, and identify and describe networks that apply to each network type.
- Understand the basic concepts of the Physical Layer, including physical media, encoding/modulation, multiplexing, error control, and their implementation in various commercial networks.
- Describe the basic operation of the Data Link Layer, including connection oriented versus connectionless protocols, retransmission algorithms, windows and flow control, and their implementations in various networks.
- Describe the basic operation of the network layer, including addressing and routing.
- Describe the basic operation of TCP/UDP, including connection establishment and release, buffered transfer, adaptive retransmission, and congestion and flow control.
- Describe LAN architectures and their implementations.
- Introduce Application layer concepts, including commercial Internet protocols and client-server technologies.
- Introduce special issues, including security, performance, and quality of service from a technical and ethical viewpoint.
- Tie in all above concepts to describe the global data telecommunications network.

Relationship between Course Outcomes and Program Outcomes
The following Program Outcomes are supported by the above Course Outcomes:

- a. An ability to apply knowledge of computing and mathematics appropriate to the discipline
- c. An ability to design, implement and evaluate a computer-based system, process, component, or program to meet desired needs
- i. An ability to use current techniques, skills, and tools necessary for computing practices.
- j. An ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices

Prerequisites by Topic

- CS455 is a senior course in Computer Science and as such expects from its students a reasonable level of mathematical and computing sophistication.
- Physical phenomena such as electrical signals are discussed but no background beyond high school physics is assumed.
- Discussion of the software aspects of data communications assumes a knowledge of: operating systems, data structures, and the organization of reasonably complicated programs.

Major Topics Covered in the Course

<table>
<thead>
<tr>
<th>Topic</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Introduction to the course, layered protocols, and networks</td>
<td>4 hours</td>
</tr>
<tr>
<td>2. Physical layer</td>
<td>5 hours</td>
</tr>
<tr>
<td>3. LANs and Medium Access Control</td>
<td>5 hours</td>
</tr>
<tr>
<td>4. Data link layer</td>
<td>4 hours</td>
</tr>
<tr>
<td>5. Network layer (IP)</td>
<td>5 hours</td>
</tr>
<tr>
<td>6. Transport layer (TCP, UDP)</td>
<td>5 hours</td>
</tr>
<tr>
<td>7. Application layer</td>
<td>4 hours</td>
</tr>
<tr>
<td>8. Special issues</td>
<td>1 hour</td>
</tr>
<tr>
<td>9. A Complete Network Overview</td>
<td>3 hours</td>
</tr>
<tr>
<td>Midterm (Review, Test), Paper / Project(s) Description &amp; Evaluation, Final Exam Review</td>
<td>9 hours</td>
</tr>
<tr>
<td>Final Exam</td>
<td>45 hours</td>
</tr>
</tbody>
</table>

Assessment Plan for the Course

End of every semester Course Objective Assessments by CS department. End of semester Course Evaluations by IIT. Reviewed every Spring semester by CS Undergraduate Studies Committee for possible updates in the following Fall. Once every 4-5 years a detailed review of all materials for the course is made by the CS Undergraduate Studies Committee.

How Data in the Course is Used to Assess Program Outcomes (unless adequately covered already in the assessment discussion under Criterion 4)

See the assessment discussion under Criterion 4

For a computer science program
<table>
<thead>
<tr>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithms</td>
<td></td>
<td></td>
<td>Software design</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Data structures</td>
<td></td>
<td></td>
<td>Concepts of programming languages</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
COURSE DESCRIPTION

<table>
<thead>
<tr>
<th>Dept., Number</th>
<th>Course Title</th>
<th>Information Security</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS458</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Semester hours: 3

Course Coordinator: Dr. David Grossman, Associate Professor

Current Catalog Description

An introduction to the fundamentals of computer and information security. This course focuses on algorithms and techniques used to defend against malicious software. Topics include an introduction to encryption systems, operating system security, database security, network security, system threats, and risk avoidance procedures. Prerequisites: CS 425 and CS 450. (3-0-3)

Textbook


References


Course Outcomes

Students should be able to:

- Provide an introduction to the security engineering discipline
- Expose students to contemporary risks and attack procedures.
- To provide students with an appreciation of the historical perspective in information assurance research.
- Describe security-engineering processes – particularly those being used in industry.
- Students will be familiar with fundamental encryption algorithms
- Students will be able to design an architecture to defend a specific system from attack.
- The student will be able to apply standard, accepted security engineering techniques to protect a system with respect to a specific organizational security policy.
- The student will demonstrate an ability to document their work to an acceptable standard.

Relationship between Course Outcomes and Program Outcomes

The following Program Outcomes are supported by the above Course Outcomes:

- c. An ability to design, implement and evaluate a computer-based system, process, component, or program to meet desired needs
- e. An understanding of professional, ethical, legal, security, and social issues and responsibilities
- f. An ability to communicate effectively with a range of audiences.
- g. An ability to analyze the local and global impact of computing on individuals, organizations and society
- i. An ability to use current techniques, skills, and tools necessary for computing practices.
- j. An ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices

### Prerequisites by Topic

Operating Systems, Databases and Programming Knowledge

### Major Topics Covered in the Course

<table>
<thead>
<tr>
<th>Topic</th>
<th>Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Security Engineering Perspectives</td>
<td>3</td>
</tr>
<tr>
<td>2. Security Historical Perspectives</td>
<td>3</td>
</tr>
<tr>
<td>3. Operating System Security</td>
<td>4.5</td>
</tr>
<tr>
<td>4. Database Security Algorithms</td>
<td>4.5</td>
</tr>
<tr>
<td>5. Network Security</td>
<td>4.5</td>
</tr>
<tr>
<td>6. Security Administration</td>
<td>4.5</td>
</tr>
<tr>
<td>7. E-Commerce Security</td>
<td>4.5</td>
</tr>
<tr>
<td>8. Encryption types and techniques</td>
<td>6</td>
</tr>
<tr>
<td>9. Prevention, Detection, and Response</td>
<td>6</td>
</tr>
<tr>
<td>10. Legal and Ethical Issues</td>
<td>4.5</td>
</tr>
</tbody>
</table>

45 hours

### Assessment Plan for the Course

End of every semester Course Objective Assessments by CS department. End of semester Course Evaluations by IIT. Reviewed every Spring semester by CS Undergraduate Studies Committee for possible updates in the following Fall. Once every 4-5 years a detailed review of all materials for the course is made by the CS Undergraduate Studies Committee.

### How Data in the Course is Used to Assess Program Outcomes (unless adequately covered already in the assessment discussion under Criterion 4)

See the assessment discussion under Criterion 4

---

### Estimate Curriculum Category Content (Semester hours)

<table>
<thead>
<tr>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithms</td>
<td></td>
<td>1.5</td>
<td>Software design</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Data structures</td>
<td></td>
<td></td>
<td>Concepts of programming languages</td>
<td>1.5</td>
<td></td>
</tr>
</tbody>
</table>
COURSE DESCRIPTION

<table>
<thead>
<tr>
<th>Dept., Number</th>
<th>CS470</th>
<th>Course Title</th>
<th>Computer Architecture</th>
</tr>
</thead>
<tbody>
<tr>
<td>Semester hours</td>
<td>3</td>
<td>Course Coordinator</td>
<td>Virgil Bistriceanu, Instructor</td>
</tr>
</tbody>
</table>

Current Catalog Description

Introduction to the functional elements and structures of digital computers. Detailed study of specific machines at the register transfer level illustrates arithmetic, memory, I/O, and instruction processing. Prerequisites: CS 350 and ECE 218. (2-2-3) (T) (C)

Textbook


References

See www.cs.iit.edu/~cs470

Course Outcomes

Students should be able to:

- Present the milestones of computer architecture history
- Fundamentals of computer design
  - Explain the difference between various measure of performance: Latency, throughput; MIPS, MPFLOS
  - Comparing performance
  - Utilize Amdahl’s law to estimate the overall speedup
  - Explain the difference between a good and a bad benchmark
- Assembly level machine organization
  - Explain the basic organization of the classical von Neumann machine and its major functional units
  - Explain how an instruction is executed in a classical von Neumann machine
  - Summarize how instructions are represented at both the machine level and in the context of a symbolic assembler
  - Explain different Instruction Set formats (0 (stack), 1 (accumulator), 2, and 3-addresses per instruction; Variable length vs. fixed length formats)
  - Design the Instruction Set for a general purpose CPU
  - Explain how the basic addressing modes work: Register, Memory direct, Memory indirect, Base and displacement, Indexed
  - Explain how base and displacement addressing is used in block-based programming languages
  - Write small MIPS assembly language programs
• Demonstrate how fundamental high-level programming constructs are implemented at the machine-language level: If-then-else, Loops (for, while, do-unti), Procedure call/return
  • Explain the basic concepts of interrupts and I/O operations

  • Datapath and Control
    • Design a single clock-cycle datapath for a CPU
    • Explain why a single clock-cycle datapath is inefficient
    • Re-factor a single clock-cycle datapath into a multi clock-cycle one
    • Explain the difference between a hardwired and a microprogrammed control unit
    • Design the control unit for a single clock-cycle datapath
    • Explain how exceptions impact the design and performance of a datapath

  • Pipelining
    • Derive the formula for the throughput of an ideal pipeline with N stages
    • Explain the limiting factors in building a pipeline with too many stages
    • Explain how data and control hazards occur and how their impact can be eliminated or reduced
    • Re-factor MIPS code to reduce/eliminate data and branch hazards
    • Explain the significance of a late commit in the pipeline
    • Explain the changes in the design and implementation of a pipelined datapath to account for exceptions
    • Explain branch prediction
    • Solve problems that require finding the real CPI of a program running on a pipelined datapath

  • The memory hierarchy
    • Identify the main types of memory technology and explain the trade-off in using them
    • Explain the effect of memory latency on running time
    • Explain the use of memory hierarchy to reduce the effective memory latency
    • Explain the differences between different cache organizations: Direct mapped, Set associative Fully associative
    • Utilize a cache simulator and access traces to compare the performance of caches with different sizes and organizations
    • Explain main memory organization alternatives to improve performance: Wide-memory, Interleaving
    • Explain the impact of access stride to performance
    • Explain the virtual memory structure and mapping
    • Explain why and how virtual memory impacts performance and how performance can be improved. TLB
    • Analyze the differences between cache organizations in systems with virtual memory: Real address caches, Pipelined real caches, Virtual address cache, Restricted virtual caches, TLB addressing

  • I/O
    • Define the meaning of various I/O performance measures
    • Types and characteristics of I/O devices
    • Explain the differences between major buses (IDE, SCSI, USB, PCI): synchronous v. asynchronous, Serial v. parallel, Number of devices, Termination,
Transfer rates
  o Design issues related to I/O system addressing: Memory-mapped I/O, Cache coherency, Snoopy controllers, DMA I/O configurations
  o Explain the sources of latency in a I/O subsystem

Relationship between Course Outcomes and Program Outcomes
The following Program Outcomes are supported by the above Course Outcomes:
  • a. An ability to apply knowledge of computing and mathematics appropriate to the discipline
  • c. An ability to design, implement and evaluate a computer-based system, process, component, or program to meet desired needs
  • f. An ability to communicate effectively with a range of audiences.
  • j. An ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices

Prerequisites by Topic
Basic understanding of a von-Neumann computer organization
The ability to explain the differences between a high level instruction and a compiled instruction
Knowledge of the steps involved in the execution of an instruction
Solid understanding of basic building blocks for a datapath: ALU, register, counter, multiplexer, decoder, glue logic
Working knowledge of Boolean logic

Major Topics Covered in the Course
1. Overview and history of computer architecture 1 hour
2. Fundamentals of computer design 3 hours
3. Basic organization of a von Neumann computer 1 hours
4. Instruction Set design 3 hours
5. Datapath and Control 4 hours
6. Pipelining 5 hours
7. The memory hierarchy 4 hours
8. I/O 4 hours
Introduction: discuss class structure, objectives, and requirements, Midterm 3 hours
Project presentation 2 hours
Laboratory 30 hours
Final Exam -
Total 60 hours

Assessment Plan for the Course
End of every semester Course Objective Assessments by CS department. End of semester Course Evaluations by IIT. Reviewed every Spring semester by CS Undergraduate Studies Committee for possible updates in the following Fall. Once every 4-5 years a detailed review of
all materials for the course is made by the CS Undergraduate Studies Committee.

How Data in the Course is Used to Assess Program Outcomes (unless adequately covered already in the assessment discussion under Criterion 4)

See the assessment discussion under Criterion 4

For a computer science program
Estimate Curriculum Category Content (Semester hours)

<table>
<thead>
<tr>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithms</td>
<td></td>
<td>1.5</td>
<td>Software design</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Data structures</td>
<td></td>
<td></td>
<td>Concepts of programming languages</td>
<td>1.5</td>
<td></td>
</tr>
</tbody>
</table>
COURSE DESCRIPTION

<table>
<thead>
<tr>
<th>Dept., Number</th>
<th>CS480</th>
<th>Course Title</th>
<th>Artificial Intelligence: Planning and Control</th>
</tr>
</thead>
<tbody>
<tr>
<td>Semester hours</td>
<td>3</td>
<td>Course Coordinator</td>
<td>Dr. Shlomo Argamon, Associate Professor</td>
</tr>
</tbody>
</table>

Current Catalog Description

Introduction to computational methods of intelligent control of autonomous agents, and the use of programming paradigms that support development of flexible and reactive systems. These include heuristic search, knowledge representation, constraint satisfaction, probabilistic reasoning, decision-theoretic control, and sensor interpretation. Particular focus will be places on real-world application of the material. (3-0-3). Prerequisite: CS 331 or CS 401 or CS 403. Corequisite: MATH 474 or equivalent. (3-0-3) (T)

Textbook


References

LISP References - textbook WWW page http://www.cs.berkeley.edu/~russell/aima.html

Course Outcomes

Students should be able to:
- Describe the Turing test.
- Explain the concepts of optimal reasoning, human-like reasoning, optimal behavior, and human-like behavior.
- Develop "PAGE" descriptions of an agents and determine which agent type is applicable to a problem.
- Solve problems in a functional programming language (LISP)
- Formulate an efficient problem space for a problem expressed in English by expressing that problem space in terms of states, operators, an initial state, and a description of a goal state.
- Describe the problem of combinatorial explosion and its consequences.
- Select an appropriate brute-force search algorithm for a problem, implement it, and characterize its time and space complexities.
- Select an appropriate heuristic search algorithm for a problem and implement it by designing the necessary heuristic evaluation function.
- Describe under what conditions heuristic algorithms guarantee optimal solution.
- Implement minimax search with alpha-beta pruning for some two-player game.
- Formulate a problem specified in English as a constraint-satisfaction problem and implement it using a chronological backtracking algorithm.
- Explain the operation of the resolution technique for theorem proving.
• Apply Bayes theorem to determine conditional probabilities.
• Explain the distinction between monotonic and non-monotonic inference.
• Explain the differences among the three main styles of learning: supervised, reinforcement, and unsupervised.
• Implement simple algorithms for supervised learning, reinforcement learning, and unsupervised learning.
• Determine which of the three learning styles is appropriate to a particular problem domain.
• Compare and contrast each of the following techniques, providing examples of when each strategy is superior: decision trees, neural networks, and belief networks. Explain the nearest neighbor algorithm and its place within learning theory.

Relationship between Course Outcomes and Program Outcomes
The following Program Outcomes are supported by the above Course Outcomes:
• a. An ability to apply knowledge of computing and mathematics appropriate to the discipline
  i. An ability to use current techniques, skills, and tools necessary for computing practices.
• j. An ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices

Prerequisites by Topic
• Programming including recursion
• Discrete mathematics and data structures
• Basic analysis of algorithms

Major Topics Covered in the Course
1. Introduction, History of AI, Intelligent agents 3 hours
2. Functional Programming (LISP) 7.5 hours
3. Uninformed search, Informed search, Constraint satisfaction, Game-playing 12 hours
4. Logical agents, Propositional logic, First-order logic, Inference in first-order logic 4.5 hours
5. Uncertainty, Probability, Belief networks, Belief network inference, Optimal decisions under uncertainty, Optimal sequential decisions 10.5 hours
6. Learning, Neural networks, Bayesian learning 6 hours
Midterm Exam 1.5 hours
Final Exam - 45 hours

Assessment Plan for the Course
End of every semester Course Objective Assessments by CS department. End of semester Course Evaluations by IIT. Reviewed every Spring semester by CS Undergraduate Studies Committee for possible updates in the following Fall. Once every 4-5 years a detailed review of all materials for the course is made by the CS Undergraduate Studies Committee.
How Data in the Course is Used to Assess Program Outcomes (unless adequately covered already in the assessment discussion under Criterion 4)

See the assessment discussion under Criterion 4

For a computer science program
Estimate Curriculum Category Content (Semester hours)

<table>
<thead>
<tr>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithms</td>
<td></td>
<td></td>
<td>Software design</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Data structures</td>
<td></td>
<td></td>
<td>Concepts of</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>programming</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>languages</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
COURSE DESCRIPTION

<table>
<thead>
<tr>
<th>Dept., Number</th>
<th>Course Title</th>
<th>Artificial Intelligence: Language Understanding</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS481</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Semester hours 3

Course Coordinator Dr. Shlomo Argamon, Associate Professor

Current Catalog Description

Theory and programming paradigms that enable systems to understand human language texts and extract useful information and knowledge. For example, extraction of structured event representations from news stories or discovering new research hypotheses by analyzing thousands of medical research articles. The course covers a variety of text analysis and text mining methods, with an emphasis on building working systems. Connections to information retrieval, data mining, and speech recognition will be discussed. (3-0-3) Prerequisite: MATH474 and (CS331 or CS401 or CS403)

Textbook


Articles placed on the web site: http://courseinfo.iit.edu/

References

Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech Recognition, by Dan Jurafsky, James H. Martin, and Keith Vander Linden


The Internet Grammar of English: http://www.ucl.ac.uk/internet-grammar/home.htm

Course Outcomes

Students should be able to:

- Build systems that analyze unstructured natural language texts and extract useful information from them.
- Explain various natural language analysis methods, with a focus on hands-on experimentation and exploring real-world applications.
- Explain a variety of existing text analysis and text mining systems.
- Explain and implement the overarching text analysis task of information extraction including:
  - Part-of-speech tagging
  - Chunking
  - Named-entity recognition
• Parsing
• Co-reference analysis

• Explain and understand the application of the following algorithms and techniques:
  o Hidden markov models
  o Instance-based learning
  o Lexical similarity measures
  o Semantic frame models
  o Clustering and classification learning techniques
  o Lexical chain analysis.

Relationship between Course Outcomes and Program Outcomes

The following Program Outcomes are supported by the above Course Outcomes:

- a. An ability to apply knowledge of computing and mathematics appropriate to the discipline
- c. An ability to design, implement and evaluate a computer-based system, process, component, or program to meet desired needs
- f. An ability to communicate effectively with a range of audiences.
- j. An ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices

Prerequisites by Topic

Algorithms, Probability

Major Topics Covered in the Course

| 1. Introduction and linguistic concepts, Practical issues in text processing, Overview of applications and architectures | 3 hours |
| 2. Part-of-speech (POS) tagging | 4.5 hours |
| 3. Shallow parsing | 4.5 hours |
| 4. Link parsing | 4.5 hours |
| 5. Dependency | 4.5 hours |
| 6. Lexical semantics | 4.5 hours |
| 7. Named-Entity Recognition | 4.5 hours |
| 8. Information Extraction | 4.5 hours |
| 9. Text Summarization | 4.5 hours |
| 10. Real-World Applications and Systems | 1.5 hours |
| 11. Text Classification | 3 hours |
| Midterm Exam | 1.5 hours |
| Final Exam | - |

Assessment Plan for the Course

V - 84
End of every semester Course Objective Assessments by CS department. End of semester Course Evaluations by IIT. Reviewed every Spring semester by CS Undergraduate Studies Committee for possible updates in the following Fall. Once every 4-5 years a detailed review of all materials for the course is made by the CS Undergraduate Studies Committee.

How Data in the Course is Used to Assess Program Outcomes (unless adequately covered already in the assessment discussion under Criterion 4)

See the assessment discussion under Criterion 4

For a computer science program
Estimate Curriculum Category Content (Semester hours)

<table>
<thead>
<tr>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithms</td>
<td></td>
<td>3</td>
<td>Software design</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Data structures</td>
<td></td>
<td></td>
<td>Concepts of programming</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>languages</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
COURSE DESCRIPTION

Dept., Number | CS482 | Course Title | Information and Knowledge Management Systems
Semester hours | 3 | Course Coordinator | Dr. Shlomo Argamon, Associate Professor

Current Catalog Description

This capstone course is designed as a project course whose purpose is to enable students to see how the various algorithms and systems they have learned about in their prerequisite courses can be used in context to create useful knowledge management tools. Students in the course will be divided into groups, each of which will choose a project early in the semester whose results they will present at the end of the semester. Periods will be divided among discussion of design of information and knowledge management systems, lectures on effective project management techniques, and hands-on advising of student project group meetings. (3-0-3)Prerequisites: CS425 and two of the following courses: CS 422, CS 429 or CS 481, or instructor's consent.

Textbook

Knowledge Management in Theory and Practice by Kimiz Dalkir
The 7 Habits of Highly Effective People by Stephen Covey

References

The Complete Idiot's Guide to Knowledge Management by Melissie Clemmons Rumizen

Course Outcomes

Students should be able to:
- Understand the goals and methods of information and knowledge management
- Describe different types of IKMS strategies
- Understand different aspects of organizational culture
- Describe how computational tools can enhance or detract from organizational effectiveness
- Describe different models for measuring IKMS effectiveness
- Implement useful computational tools for supporting collaborative knowledge work
- Critique the effectiveness of different user-interfaces
- Describe how intellectual property issues affect the development and deployment of software systems
- Describe how personal interaction styles can affect team effectiveness
- Develop a personal vision and associated goals
- Develop clear plans to accomplish such goals
- Know how to manage their own time and resources effectively
- Understand how to negotiate win/win agreements
Create plans for self-improvement in a professional context

Relationship between Course Outcomes and Program Outcomes

The following Program Outcomes are supported by the above Course Outcomes:

- a. An ability to apply knowledge of computing and mathematics appropriate to the discipline
- c. An ability to design, implement and evaluate a computer-based system, process, component, or program to meet desired needs
- d. An ability to function effectively on teams to accomplish a common goal
- f. An ability to communicate effectively with a range of audiences.
- g. An ability to analyze the local and global impact of computing on individuals, organizations and society
- i. An ability to use current techniques, skills, and tools necessary for computing practices.
- j. An ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices

Prerequisites by Topic

CS425 (Databases) and any two of:
CS422 (Data Mining), CS429 (Information Retrieval), CS481 (Artificial Intelligence: Natural Language Processing)

Major Topics Covered in the Course

1. Introduction to Knowledge Management & IKMS 3 hours
2. Project Management 4 hours
3. KM Strategies and case studies 4 hours
4. Organizational culture & Communities of practice 4 hours
5. Intellectual Property issues 3 hours
6. Teamwork and interactional styles 3 hours
7. Business issues in IT 4 hours
8. Data Warehousing 3 hours
9. Measuring Knowledge Management 3 hours
10. Human Computer Interaction 4 hours
11. Personal Management Skills 10 hours

Final Exam - 45 hours

Assessment Plan for the Course

End of every semester Course Objective Assessments by CS department. End of semester Course Evaluations by IIT. Reviewed every Spring semester by CS Undergraduate Studies Committee for possible updates in the following Fall. Once every 4-5 years a detailed review of all materials for the course is made by the CS Undergraduate Studies Committee.
How Data in the Course is Used to Assess Program Outcomes (unless adequately covered already in the assessment discussion under Criterion 4)

See the assessment discussion under Criterion 4

For a computer science program

Estimate Curriculum Category Content (Semester hours)

<table>
<thead>
<tr>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithms</td>
<td></td>
<td>1.5</td>
<td>Software design</td>
<td></td>
<td>1.5</td>
</tr>
<tr>
<td>Data structures</td>
<td></td>
<td></td>
<td>Concepts of</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>programming</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>languages</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
6. Faculty

Criterion Part A: Faculty Qualifications - Criterion

Faculty members teaching in the program are current and active in the associated computing discipline. They each have the educational background and expertise consistent with their expected contributions to the program. Each has a level of competence obtained through graduate work in the discipline, relevant experience, or relevant scholarship. Collectively, they have the technical breadth and depth necessary to support the program.

For Computer Science Programs:
Most full time faculty members have a Ph.D. in computer science.

For Information Systems Programs:
Some full-time faculty, including those responsible for the IS curriculum development, hold a terminal degree in information systems.

Criterion Part B: Faculty Size and Workload - Criterion

There are enough full time faculty members to provide continuity, oversight and stability, to cover the curriculum, and to allow an appropriate mix of teaching, professional development, scholarly activities, and service for each faculty member. The faculty assigned to the program has appropriate authority for the creation, delivery, evaluation and modification of the program, and the responsibility for the consistency and quality of its courses.

Faculty Overview
We provide a detailed summary of each faculty member’s experience. We note that every full-time tenure-track (Track I) faculty member holds a Ph.D. in Computer Science or Computer Engineering. Most (six out of seven) full-time non-tenure (Track II - Industry Associate, Industry Professor, Clinical Associate Professor, Senior Lecturer) faculty member holds a Ph.D. in Computer Science. All five full-time instructors hold an M.S. in Computer Science.

In 1998, the university aggressively decided to improve quality of the Computer Science department. As such, an endowed chair position: the IITRI Professor of Computer Science was established. Professor Ophir Frieder was hired. Professor Frieder is Fellow of the AAAS, ACM, and IEEE. He is the 2007 American Society of Information Science and Technology (ASIS&T) Research Award recipient. He has published over 150 publications. He is on the steering committee of a number of conferences. He was immediately put in charge of the Computer Science faculty search committee. Within a few years of his appointment at IIT, eleven tenured-track faculty members were hired, about half of them since the last ABET accreditation visit. The name, school where they earned their Ph.D or where they were working when they were hired and the year hired as well as their area of expertise is provided below.
• Professor Xian-He Sun, tenured associate professor at Louisiana State University, 1999, Systems Architecture, Operating Systems
• Associate Professor: Gruia Calinescu, Georgia Tech, 1999, Algorithms
• Associate Professor, Gady Agam, Concordia, 2000, Image Processing
• Associate Professor, Xiang-Yang Li, UIUC, 2000, Network Algorithms
• Professor, Ed Reingold, tenured professor at UIUC, 2000, Algorithms
• Professor Sanjiv Kapoor, tenured professor at IIT India, 2001, Algorithms
• Assistant Professor, Zhilin Lan, Northwestern University, 2002, Systems Architecture, Operating Systems
• Associate Professor, Shlomo Argamon, faculty member at Bar-Ilan University, 2002, Computational Linguistics, Artificial Intelligence
• Assistant Professor, Wai Gen Yee, Georgia Tech, 2003, Database Systems
• Assistant Professor, Shangping Ren, Tellabs, 2003, Programming Languages

Some of these new faculty have already demonstrated outstanding research excellence. Dr. Wan who now has over 60 major journal publications and the department is proud to have two NSF Career Grant recipients in Dr. Hood and Dr. Ren. With a department of only sixteen faculty members, the hiring of eleven new faculty members in the last ten years has dramatically improved the quality of the department. Each of these hires has established a substantial resume, and each has a history of excellence in teaching as well as research. The ability to recruit candidates from top programs from around the world is now relatively straightforward and something that simply could not have easily been done in the 1990’s.

It should also be noted that these new faculty have brought about real change in the undergraduate curriculum. The Undergraduate Studies Committee was formed in 1999 and Professor Grossman was named the chair of the committee. At present, Professors Argamon, Yee, and Lan are the tenure-track committee members, Professor Goharian, the three time Computer Science Department Teacher of the Year guarantees quality of teaching, and the department’s number one expert on undergraduate curriculum, Professor Bauer comprise the committee. Professor Bauer has a very deep understanding of undergraduate advising challenges, as he has been the key undergraduate advisor in our department for fifteen years. These faculty members are clearly able to “to provide continuity, oversight and stability.”

Given the broad list of specialties of these new hires, they provide broad coverage of our curriculum. Coupled with the existing faculty and extremely qualified instructors (several hold a Ph.D. in Computer Science) we believe they are able to cover the curriculum reasonably.

Professor Frieder also led the department chair search committee, which resulted in the appointment of Professor Ed Reingold as the Department Chair from 2001-2006. Professor Reingold is an internationally recognized expert in algorithms as he has published multiple programming, data structure and algorithm textbooks.
Departmental resources, as a direct result of research funding, have dramatically improved as well with the addition of these faculty members. The IIT Information Retrieval lab led by Professors Frieder and Grossman has brought in over $4.5M in research funding since its establishment in 1999. This has led to resources available to develop new courses in information retrieval, data mining and knowledge management. Recent funding to the IIT Information Security lab led to the development of a new course (much enjoyed by our undergrads as seen by large enrollments and positive evaluations) on Information Security.

It should be noted that Prof. Korel is now the interim department chair. At first review, it may appear that the lack of a named Department Chair would be a major problem for the department. The truth is that the renovation of department faculty has dramatically raised the standards for the chair search committee. Hence, numerous applicants have been interviewed but none have met the high standards of the committee. Additionally, the turnover in the administration (new president, and soon, a new provost and a new dean) clearly affects the ability to attract top candidates as they obviously wish to know more about the details of their management chain. Certainly, the department is eager to settle the position of department chair, but Prof. Korel has done an outstanding job as acting chair since 2006 as can be seen by solid course evaluations, consistently student evaluations, alumni surveys and a steady stream of resources generated by research funding.

A. Faculty Profile

1. Please complete the following table for each faculty member who regularly teaches courses in the program.

<table>
<thead>
<tr>
<th>Faculty Name</th>
<th>Rank</th>
<th>FT/PT</th>
<th>Highest Deg./Field</th>
<th>Research Areas</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bauer C.</td>
<td>Professor</td>
<td>PT</td>
<td>M.S. Math</td>
<td>Education</td>
</tr>
<tr>
<td></td>
<td>(Emeritus)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Carlson</td>
<td>Professor</td>
<td>FT</td>
<td>Ph.D. CS</td>
<td>Information Systems</td>
</tr>
<tr>
<td>Frieder</td>
<td>Professor</td>
<td>FT</td>
<td>Ph.D. CS</td>
<td>Information Systems</td>
</tr>
<tr>
<td>Kapoor</td>
<td>Professor</td>
<td>FT</td>
<td>Ph.D. CS</td>
<td>Algorithms</td>
</tr>
<tr>
<td>Reingold</td>
<td>Professor</td>
<td>FT</td>
<td>Ph.D. CS</td>
<td>Algorithms</td>
</tr>
<tr>
<td>Sun</td>
<td>Professor</td>
<td>FT</td>
<td>Ph.D. CS</td>
<td>Operating Systems</td>
</tr>
<tr>
<td>Agam</td>
<td>Associate</td>
<td>FT</td>
<td>Ph.D. CS</td>
<td>Computer Graphics</td>
</tr>
<tr>
<td></td>
<td>Professor</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Argamon</td>
<td>Associate</td>
<td>FT</td>
<td>Ph.D. CS</td>
<td>Artificial Intelligence</td>
</tr>
<tr>
<td></td>
<td>Professor</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Calinescu</td>
<td>Associate</td>
<td>FT</td>
<td>Ph.D. CS</td>
<td>Algorithms</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Name</td>
<td>Position/Lecturer</td>
<td>Type</td>
<td>Degree</td>
<td>Department</td>
</tr>
<tr>
<td>--------------</td>
<td>-------------------</td>
<td>------</td>
<td>--------</td>
<td>---------------------------------</td>
</tr>
<tr>
<td>Elrad</td>
<td>Associate Professor</td>
<td>PT</td>
<td>Ph.D. CS</td>
<td>Software Engineering</td>
</tr>
<tr>
<td>Grossman</td>
<td>Associate Professor</td>
<td>FT</td>
<td>Ph.D. CS</td>
<td>Information Systems</td>
</tr>
<tr>
<td>Hood</td>
<td>Associate Professor / Associate Dept Chair</td>
<td>FT</td>
<td>Ph.D. CPE</td>
<td>Computer Networking</td>
</tr>
<tr>
<td>Korel</td>
<td>Associate Professor / Acting Dept. Chair</td>
<td>FT</td>
<td>Ph.D. CS</td>
<td>Software Engineering</td>
</tr>
<tr>
<td>Li</td>
<td>Associate Professor</td>
<td>FT</td>
<td>Ph.D. CS</td>
<td>Algorithms and Computer Networking</td>
</tr>
<tr>
<td>Wan</td>
<td>Associate Professor</td>
<td>FT</td>
<td>Ph.D. CS</td>
<td>Computer Networking</td>
</tr>
<tr>
<td>Lan</td>
<td>Assistant Professor</td>
<td>FT</td>
<td>Ph.D. CS</td>
<td>Operating Systems</td>
</tr>
<tr>
<td>Ren</td>
<td>Assistant Professor</td>
<td>FT</td>
<td>Ph.D. CS</td>
<td>Programming Languages</td>
</tr>
<tr>
<td>Yee</td>
<td>Assistant Professor</td>
<td>FT</td>
<td>Ph.D. CS</td>
<td>Information Systems</td>
</tr>
<tr>
<td>Roberson</td>
<td>Research Professor</td>
<td>PT</td>
<td>M.S. EE</td>
<td>Computer Networking</td>
</tr>
<tr>
<td>Leung</td>
<td>Industry Professor</td>
<td>FT</td>
<td>Ph.D. CS</td>
<td>Operating Systems</td>
</tr>
<tr>
<td>Goharian</td>
<td>Clinical Associate Professor</td>
<td>FT</td>
<td>Ph.D. CS</td>
<td>Information Systems</td>
</tr>
<tr>
<td>Chlebus</td>
<td>Industry Associate Professor</td>
<td>FT</td>
<td>Ph.D. CS</td>
<td>Networking</td>
</tr>
<tr>
<td>Bauer M.</td>
<td>Senior Lecturer</td>
<td>FT</td>
<td>M.S. Math</td>
<td>Algorithms</td>
</tr>
<tr>
<td>Beckman</td>
<td>Senior Lecturer</td>
<td>FT</td>
<td>Ph.D. CS</td>
<td>Programming Languages</td>
</tr>
<tr>
<td>Sasaki</td>
<td>Senior</td>
<td>FT</td>
<td>Ph.D. CS</td>
<td>Programming</td>
</tr>
<tr>
<td>Lecturer</td>
<td>Role</td>
<td>FT/PT</td>
<td>Degree</td>
<td>Languages</td>
</tr>
<tr>
<td>------------</td>
<td>--------------</td>
<td>-------</td>
<td>-----------------------</td>
<td>------------------------------------</td>
</tr>
<tr>
<td>Soneru</td>
<td>Senior Lecturer</td>
<td>FT</td>
<td>Ph. D. CS</td>
<td>Computer Networking</td>
</tr>
<tr>
<td>Hanrath</td>
<td>Senior Instructor</td>
<td>FT</td>
<td>MS. CS</td>
<td>Programming Languages</td>
</tr>
<tr>
<td>Aldawud</td>
<td>Instructor</td>
<td>PT</td>
<td>Ph.D. CS</td>
<td>Software Engineering</td>
</tr>
<tr>
<td>Bader</td>
<td>Instructor</td>
<td>PT</td>
<td>Ph.D. CS</td>
<td>Software Engineering</td>
</tr>
<tr>
<td>Bistrieanu</td>
<td>Instructor</td>
<td>FT</td>
<td>M.S. EE</td>
<td>Software Engineering</td>
</tr>
<tr>
<td>Choi</td>
<td>Instructor</td>
<td>PT</td>
<td>Ph.D. CS</td>
<td>Computer Networking</td>
</tr>
<tr>
<td>Koutsogiannakis</td>
<td>Instructor</td>
<td>FT</td>
<td>M.S. Physics and CS</td>
<td>Programming Languages</td>
</tr>
<tr>
<td>Lee</td>
<td>Instructor</td>
<td>FT</td>
<td>M.S. CS</td>
<td>Information Systems</td>
</tr>
<tr>
<td>Manov</td>
<td>Instructor</td>
<td>PT</td>
<td>Ph.D. CS</td>
<td>Information Systems</td>
</tr>
<tr>
<td>North</td>
<td>Instructor</td>
<td>PT</td>
<td>Ph.D. CS</td>
<td>Computer Networking</td>
</tr>
<tr>
<td>Winans</td>
<td>Instructor</td>
<td>FT</td>
<td>M.S. CS</td>
<td>Programming Languages</td>
</tr>
<tr>
<td>Zhang</td>
<td>Instructor</td>
<td>PT</td>
<td>Ph.D. Math</td>
<td>Numerical Computing</td>
</tr>
</tbody>
</table>

**FT/PT = Full-time/Part-time**

Advising responsibilities for Ph.D. and M.S. students are distributed evenly across all tenure track faculty during the admissions process. Prof. Winans does initial advising for M.S. students, and Prof. Sasaki for Professional Masters students. Prof. Bauer is the only person designated as an advisor to undergraduates. All faculty are available for mentoring Undergraduates, and Prof. Bauer refers students to appropriate faculty based on the students subject area interest.

**B. Information Regarding Faculty Members**

On separate pages, please furnish the following information for all faculty members who teach courses allowed for the major, including those who have administrative positions in the department (chair, associate chair, etc.). Use the form given below as guidance. This form need not be followed exactly, but all requested information should be supplied. Use a common format for all faculty members. Limit information to no more than three pages per person, providing only the most recent information if needed to limit space. Place the form(s) for administrators first, followed by the others in alphabetical order.
If you are having more than one program evaluated, particularly if the programs are on separate campuses, indicate clearly the program(s) and/or campus(es) to which an individual is assigned, and the percentage of time to each, if more than one.
Professor
1. Name, current academic rank, and tenure status:

<table>
<thead>
<tr>
<th>Name:</th>
<th>Charles R. Bauer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rank:</td>
<td>Professor Emeritus</td>
</tr>
<tr>
<td>Tenure Status:</td>
<td>N/A</td>
</tr>
</tbody>
</table>

2. Date of original appointment to this faculty, followed by dates and ranks of advancement:

<table>
<thead>
<tr>
<th>Title</th>
<th>Dates Held</th>
</tr>
</thead>
<tbody>
<tr>
<td>Adjunct Associate Professor</td>
<td>1964-1985</td>
</tr>
<tr>
<td>Associate Professor</td>
<td>1985-1996</td>
</tr>
<tr>
<td>Emeritus Professor</td>
<td>1996-Present</td>
</tr>
</tbody>
</table>

3. Degrees with fields, institutions, and dates

<table>
<thead>
<tr>
<th>Degree</th>
<th>Field</th>
<th>Institution</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>BS</td>
<td>MATH</td>
<td>LOYOLA UNIVERSITY, CHICAGO</td>
<td>1952</td>
</tr>
<tr>
<td>M Ed</td>
<td>Education</td>
<td>LOYOLA UNIVERSITY, CHICAGO</td>
<td>1956</td>
</tr>
</tbody>
</table>

4. If you do not have an advanced degree in the program area, describe any course work you may have taken, or other ways in which you have achieved competence in the program area; there is no necessity to repeat information here that is contained in later sections of this document.

5. Conferences, workshops, and professional development programs in which you have participated in the last 5 years to improve teaching and professional competence in the program area:


6. Other related computing experience including teaching, industrial, governmental, etc. (where, when, description and scope of duties):

<table>
<thead>
<tr>
<th>Dates</th>
<th>Where</th>
<th>Duties</th>
</tr>
</thead>
</table>

VI - 8
7. Consulting—list agencies and dates, and briefly describe each project:

<table>
<thead>
<tr>
<th>Dates</th>
<th>Agency</th>
<th>Project</th>
</tr>
</thead>
</table>

8. For the academic year in which the Self Study was written list your assigned duties other than for teaching, (committee membership, advising, etc.) with average hours per week. Indicate which, if any, carry extra compensation. If you are course coordinator for courses taught by other than full-time or part time faculty, please indicate here which courses.

Department scheduling of classes

9. For the four years preceding the Self Study, list all department, college, and/or university committees of which you are/were a member including year(s) served:

University class scheduling committee

10. Principal publications during the last five years. Give in standard bibliographic format.

11. Other scholarly activity during the last 5 years: grants, sabbaticals, software development, etc.: 

12. Courses taught this and last academic year term-by-term. This year is the academic year in which this Self-Study report is prepared; the last year was the year prior to this. If you were on sabbatical leave, please enter the information for the year prior to the sabbatical. Please list each section of the same course separately.

<table>
<thead>
<tr>
<th>Term/year</th>
<th>Course number</th>
<th>Course title</th>
<th>Semester hrs</th>
<th>Class size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fall 2006</td>
<td>CS 485</td>
<td>Computers and Society</td>
<td>3</td>
<td>17</td>
</tr>
<tr>
<td>Fall 2006</td>
<td>CS 560</td>
<td>Computer Science in the Classroom</td>
<td>3</td>
<td>17</td>
</tr>
<tr>
<td>Spring 2007</td>
<td>CS 485</td>
<td>Computers and Society</td>
<td>3</td>
<td>23</td>
</tr>
<tr>
<td>Spring 2007</td>
<td>CS 561</td>
<td>The Computer and Curriculum Content</td>
<td>3</td>
<td>14</td>
</tr>
<tr>
<td>Fall 2007</td>
<td>CS 485</td>
<td>Computers and Society</td>
<td>3</td>
<td>21</td>
</tr>
<tr>
<td>Fall 2007</td>
<td>CS 560</td>
<td>Computer Science in the Classroom</td>
<td>3</td>
<td>13</td>
</tr>
<tr>
<td>Spring 2007</td>
<td>CS 561</td>
<td>The Computer and Curriculum Content</td>
<td>3</td>
<td>14</td>
</tr>
</tbody>
</table>

13. Estimate the percentage of your time devoted to scholarly and/or research activities: 25%. Please give a brief description of your major research and scholarly activities:

Training of graduate students to prepare them for future teaching.

14. If you are a part time faculty member or a full-time faculty member without full-time commitment to the program, state what percentage of full-time you are assigned to the program: ____%.
1. Name, current academic rank, and tenure status:

<table>
<thead>
<tr>
<th></th>
<th>C. Robert Carlson</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name:</td>
<td></td>
</tr>
<tr>
<td>Rank:</td>
<td>Professor</td>
</tr>
<tr>
<td>Tenure Status:</td>
<td>Tenured</td>
</tr>
</tbody>
</table>

2. Date of original appointment to this faculty, followed by dates and ranks of advancement:

<table>
<thead>
<tr>
<th>Title</th>
<th>Dates Held</th>
</tr>
</thead>
<tbody>
<tr>
<td>Associate Professor and Chair</td>
<td>August 1984</td>
</tr>
<tr>
<td>Professor</td>
<td>August 1989</td>
</tr>
</tbody>
</table>

3. Degrees with fields, institutions, and dates

<table>
<thead>
<tr>
<th>Degree</th>
<th>Field</th>
<th>Institution</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>BA BA</td>
<td>Math Accounting</td>
<td>Augustana College</td>
<td>June 1966</td>
</tr>
<tr>
<td>MS</td>
<td>Computer Science</td>
<td>University of Iowa</td>
<td>June 1968</td>
</tr>
<tr>
<td>PhD</td>
<td>Computer Science</td>
<td>University of Iowa</td>
<td>August 1972</td>
</tr>
</tbody>
</table>

4. If you do not have an advanced degree in the program area, describe any course work you may have taken, or other ways in which you have achieved competence in the program area; there is no necessity to repeat information here that is contained in later sections of this document.

5. Conferences, workshops, and professional development programs in which you have participated in the last 5 years to improve teaching and professional competence in the program area:

- Netsecure Conference, General Chair for each of the last six years.
- VoIP Conference, General Chair for each of the last three years.
- Motorola Innovation Workshop, Invited Speaker
6. Other related computing experience including teaching, industrial, governmental, etc. (where, when, description and scope of duties):

<table>
<thead>
<tr>
<th>Dates</th>
<th>Where</th>
<th>Duties</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

7. Consulting—list agencies and dates, and briefly describe each project:

<table>
<thead>
<tr>
<th>Dates</th>
<th>Agency</th>
<th>Project</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

8. For the academic year in which the Self Study was written list your assigned duties other than for teaching, (committee membership, advising, etc.) with average hours per week. Indicate which, if any, carry extra compensation. If you are course coordinator for courses taught by other than full-time or part time faculty, please indicate here which courses.

Director Rice Campus – coordinate course offerings on this campus from various IIT colleges -- 10 hours per week
Director Center for Professional Development – director of four degree programs and both the domestic and international professional learning programs – 40 hours per week
Associate Dean Graduate College – graduate college outreach activities – 10 hours per week

9. For the four years preceding the Self Study, list all department, college, and/or university committees of which you are/were a member including year(s) served:

University -- Graduate Studies Committee, 2003 to present
University – Graduation Speaker Committee, 2003 to 2007
Chair Search Committee Chemical Engineering 2003

10. Principal publications during the last five years. Give in standard bibliographic format.

Sargon Hasso and C. R. Carlson, Linguistic-based Design Pattern Classification, HICSS
11. Other scholarly activity during the last 5 years: grants, sabbaticals, software development, etc.:

<table>
<thead>
<tr>
<th>Year</th>
<th>Grant Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2002</td>
<td>State of Illinois, $54,000, PI WESTEC West Suburban Technology Enterprise Center</td>
</tr>
<tr>
<td>2002</td>
<td>Manufacturer’s Association Grant (with Keith McKee) $24,000</td>
</tr>
<tr>
<td>2003</td>
<td>State of Illinois, $54,000, WESTEC West Suburban Technology Enterprise Center</td>
</tr>
<tr>
<td>2003</td>
<td>Manufacturer’s Association Grant (with Keith McKee) $12,000</td>
</tr>
<tr>
<td>2003</td>
<td>Rice Foundation Grant, PI, $5,000</td>
</tr>
<tr>
<td>2003</td>
<td>Elkay Manufacturing Grant, $2,000</td>
</tr>
<tr>
<td>2004</td>
<td>State of Illinois, $37,500, WESTEC West Suburban Enterprise Center</td>
</tr>
<tr>
<td>2004</td>
<td>Elkay Manufacturing Grant, $2,000</td>
</tr>
<tr>
<td>2004</td>
<td>Lucent $13,000 equipment grant (with Carol Davids)</td>
</tr>
<tr>
<td>2005</td>
<td>State of Illinois, $5,000, WESTEC West Suburban Enterprise Center</td>
</tr>
<tr>
<td>2005</td>
<td>Elkay Manufacturing Grant, $2,000</td>
</tr>
<tr>
<td>2006</td>
<td>Elkay Manufacturing Grant, $2,000</td>
</tr>
<tr>
<td>2007</td>
<td>Motorola, Foundation, $25,000,</td>
</tr>
<tr>
<td>2007</td>
<td>Elkay Manufacturing Grant, $2,000</td>
</tr>
<tr>
<td>2007</td>
<td>Lisle Technology Partners Grant, $5,000</td>
</tr>
</tbody>
</table>

12. Courses taught this and last academic year term-by-term. This year is the academic year in which this Self-Study report is prepared; the last year was the year prior to this. If you were on sabbatical leave, please enter the information for the year prior to the sabbatical. Please list each section of the same course separately.

<table>
<thead>
<tr>
<th>Term/year</th>
<th>Course number</th>
<th>Course title</th>
<th>Semester hrs</th>
<th>Class size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spring 08</td>
<td>ITM 532</td>
<td>UML Based Software Development</td>
<td>3</td>
<td>42</td>
</tr>
<tr>
<td>Semester</td>
<td>Course</td>
<td>Title</td>
<td>Credits</td>
<td>Units</td>
</tr>
<tr>
<td>-----------</td>
<td>---------</td>
<td>--------------------------------------------</td>
<td>---------</td>
<td>-------</td>
</tr>
<tr>
<td>Spring 08</td>
<td>ITM 581</td>
<td>IT Entrepreneurship</td>
<td>3</td>
<td>28</td>
</tr>
<tr>
<td>Fall 07</td>
<td>IT 481</td>
<td>IT Entrepreneurship</td>
<td>3</td>
<td>17</td>
</tr>
<tr>
<td>Sum 07</td>
<td>ITM 581</td>
<td>IT Entrepreneurship</td>
<td>3</td>
<td>22</td>
</tr>
<tr>
<td>Sum 07</td>
<td>ITM 531</td>
<td>Object Oriented Sys Anal Mod/Desiign</td>
<td>3</td>
<td>10</td>
</tr>
</tbody>
</table>

13. Estimate the percentage of your time devoted to scholarly and/or research activities: __10__%. Please give a brief description of your major research and scholarly activities:
- Testing Maturity Model
- Pattern Classification Scheme
- Component based Design
- Automated Effort Metric Techniques

14. If you are a part time faculty member or a full-time faculty member without full-time commitment to the program, state what percentage of full-time you are assigned to the program: __5__%.
1. Name, current academic rank, and tenure status:

<table>
<thead>
<tr>
<th>Name:</th>
<th>Ophir Frieder</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rank:</td>
<td>Professor and IITRI Chair</td>
</tr>
<tr>
<td>Tenure Status:</td>
<td>Tenured</td>
</tr>
</tbody>
</table>

2. Date of original appointment to this faculty, followed by dates and ranks of advancement:

<table>
<thead>
<tr>
<th>Title</th>
<th>Dates Held</th>
</tr>
</thead>
<tbody>
<tr>
<td>IITRI Chair Professor of Computer Science</td>
<td>September 1998 – present</td>
</tr>
<tr>
<td>Royden B. Davis, S.J., Chair for Interdisciplinary Studies, Georgetown University</td>
<td>Sabbatical: August 2007 – present</td>
</tr>
</tbody>
</table>

3. Degrees with fields, institutions, and dates

<table>
<thead>
<tr>
<th>Degree</th>
<th>Field</th>
<th>Institution</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>BS</td>
<td>CCS</td>
<td>University of Michigan</td>
<td>84</td>
</tr>
<tr>
<td>MS</td>
<td>CSE</td>
<td>University of Michigan</td>
<td>85</td>
</tr>
<tr>
<td>PhD</td>
<td>CSE</td>
<td>University of Michigan</td>
<td>87</td>
</tr>
</tbody>
</table>

4. If you do not have an advanced degree in the program area, describe any course work you may have taken, or other ways in which you have achieved competence in the program area; there is no necessity to repeat information here that is contained in later sections of this document.

N/A

5. Conferences, workshops, and professional development programs in which you have participated in the last 5 years to improve teaching and professional competence in the program area:

- ACM SIGIR – multiple years
- ACM CIKM – multiple years
- NIST TREC – multiple years

6. Other related computing experience including teaching, industrial, governmental, etc. (where, when, description and scope of duties):

<table>
<thead>
<tr>
<th>Dates</th>
<th>Where</th>
<th>Duties</th>
</tr>
</thead>
</table>
7. Consulting—list agencies and dates, and briefly describe each project:

<table>
<thead>
<tr>
<th>Dates</th>
<th>Agency</th>
<th>Project</th>
</tr>
</thead>
<tbody>
<tr>
<td>2001-</td>
<td>Numerous Law Firms</td>
<td>Expert Witness</td>
</tr>
<tr>
<td>present</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

8. For the academic year in which the Self Study was written list your assigned duties other than for teaching, (committee membership, advising, etc.) with average hours per week. Indicate which, if any, carry extra compensation. If you are course coordinator for courses taught by other than full-time or part time faculty, please indicate here which courses.

On Sabbatical

9. For the four years preceding the Self Study, list all department, college, and/or university committees of which you are/were a member including year(s) served:

<table>
<thead>
<tr>
<th>Committee</th>
<th>Year(s) Served</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS Tenure &amp; Promotion</td>
<td>all years</td>
</tr>
<tr>
<td>University Wide Tenure &amp; Promotion</td>
<td>2003 – 2005</td>
</tr>
<tr>
<td>Chairman Review Committee</td>
<td>2004</td>
</tr>
</tbody>
</table>

10. Principal publications during the last five years. Give in standard bibliographic format.


11. Other scholarly activity during the last 5 years: grants, sabbaticals, software development, etc.:


12. Courses taught this and last academic year term-by-term. This year is the academic year in which this Self-Study report is prepared; the last year was the year prior to this. If you were on sabbatical leave, please enter the information for the year prior to the sabbatical. Please list each section of the same course separately.

<table>
<thead>
<tr>
<th>Term/year</th>
<th>Course number</th>
<th>Course title</th>
<th>Semester hrs</th>
<th>Class size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fall 06</td>
<td>CS522</td>
<td>Advanced Data Mining</td>
<td>3</td>
<td>12</td>
</tr>
<tr>
<td>Spring 07</td>
<td>CS529</td>
<td>Advanced Information Retrieval</td>
<td>3</td>
<td>11</td>
</tr>
</tbody>
</table>

13. Estimate the percentage of your time devoted to scholarly and/or research activities: **70**%. Please give a brief description of your major research and scholarly activities:

I focus on development of scalable solutions to search of structured, semi-structured and unstructured data. I am the director of the IIT Information Retrieval Lab (www.ir.iit.edu).

14. If you are a part time faculty member or a full-time faculty member without full-time commitment to the program, state what percentage of full-time you are assigned to the program: **N/A**%.
1. Name, current academic rank, and tenure status:

<table>
<thead>
<tr>
<th>Name</th>
<th>Sanjiv Kapoor</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rank:</td>
<td>Professor</td>
</tr>
<tr>
<td>Tenure Status:</td>
<td>Tenured</td>
</tr>
</tbody>
</table>

2. Date of original appointment to this faculty, followed by dates and ranks of advancement:

<table>
<thead>
<tr>
<th>Title</th>
<th>Dates Held</th>
</tr>
</thead>
<tbody>
<tr>
<td>Professor</td>
<td>August 2001-present</td>
</tr>
</tbody>
</table>

3. Degrees with fields, institutions, and dates

<table>
<thead>
<tr>
<th>Degree</th>
<th>Field</th>
<th>Institution</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>B.Tech</td>
<td>Electrical Engineering</td>
<td>Indian Institute of Technology, Delhi</td>
<td>July 1981</td>
</tr>
<tr>
<td>Ph.D</td>
<td>Computer Science</td>
<td>University of Illinois at Urbana-Champaign</td>
<td>July 1986</td>
</tr>
</tbody>
</table>

4. If you do not have an advanced degree in the program area, describe any course work you may have taken, or other ways in which you have achieved competence in the program area; there is no necessity to repeat information here that is contained in later sections of this document.

N/A

5. Conferences, workshops, and professional development programs in which you have participated in the last 5 years to improve teaching and professional competence in the program area:

- Workshop on Data Structures and Algorithms, Ottawa, Canada, 2003
- Workshop on Approximation Algorithms, Harvard University, Aug 2004
- INFORMS Annual Conference, Denver, Sept. 2004
- Foundations of Theoretical Computer Science (FSTTCS), Delhi, India, 2007.

6. Other related computing experience including teaching, industrial, governmental, etc. (where, when, description and scope of duties):

<table>
<thead>
<tr>
<th>Dates</th>
<th>Where</th>
<th>Duties</th>
</tr>
</thead>
<tbody>
<tr>
<td>May 2002</td>
<td>Max-Planck Institute for Informatik, Germany</td>
<td>Visiting Researcher</td>
</tr>
</tbody>
</table>
May 2004 IBM Research Lab, Delhi, India Visiting Researcher

7. Consulting—list agencies and dates, and briefly describe each project:
N/A

8. For the academic year in which the Self Study was written list your assigned duties other than for teaching, (committee membership, advising, etc.) with average hours per week. Indicate which, if any, carry extra compensation. If you are course coordinator for courses taught by other than full-time or part time faculty, please indicate here which courses.

<table>
<thead>
<tr>
<th>Role</th>
<th>Hours per Week</th>
</tr>
</thead>
<tbody>
<tr>
<td>Member, Graduate Committee</td>
<td>½ hr/week</td>
</tr>
<tr>
<td>Member, Campus Committee for Promotions and Tenure</td>
<td>1/2 hr (10 hrs/semester)</td>
</tr>
<tr>
<td>Member, Awards Committee, CS Department</td>
<td></td>
</tr>
<tr>
<td>Member, CS Dept. Promotion Committee</td>
<td></td>
</tr>
</tbody>
</table>

9. For the four years preceding the Self Study, list all department, college, and/or university committees of which you are/were a member including year(s) served:

<table>
<thead>
<tr>
<th>Committee</th>
<th>Years Served</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chair, Graduate Committee, CS dept</td>
<td>2002-2007</td>
</tr>
<tr>
<td>Member, University faculty Council</td>
<td>2004-2007</td>
</tr>
<tr>
<td>Member, CS Chair Search Committee</td>
<td>2005-2007</td>
</tr>
<tr>
<td>Member, CS Tenure and Promotions Committee</td>
<td>2002-2008</td>
</tr>
<tr>
<td>Member, CS awards Committee</td>
<td>2002-2008</td>
</tr>
</tbody>
</table>

10. Principal publications during the last five years. Give in standard bibliographic format.

<table>
<thead>
<tr>
<th>Publication</th>
<th>Authors</th>
<th>Conference/Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bounded Diameter Graph Problems</td>
<td>S. Kapoor and M. Sarwat</td>
<td>Theory of</td>
</tr>
</tbody>
</table>
An Auction-Based Market Equilibrium Algorithm for the Separable Gross
Substitutability Case, Rahul Garg, Sanjiv Kapoor, Vijay V. Vazirani, pp. 128-138,
APPROX-RANDOM LNCS(3122), Springer-Verlag, 2004.

Bounded Hops Power Assignment in Ad-hoc Wireless Networks," G. Calinescu, S.
Kapoor, and M. Sarwat, (Extended version in Discrete and Applied Mathematics, 2006),

Network Lifetime and Power Assignment in Ad-Hoc Wireless Networks, G. Calinescu,
S. Kapoor, A. Olshevsky and A. Zelikovsky, Proc. of European Symposium on

11. Other scholarly activity during the last 5 years: grants, sabbaticals, software
development, etc.:

Software Project: http directory server: 2008-In progress

12. Courses taught this and last academic year term-by-term. This year is the academic
year in which this Self-Study report is prepared; the last year was the year prior to
this. If you were on sabbatical leave, please enter the information for the year prior to
the sabbatical. Please list each section of the same course separately.

<table>
<thead>
<tr>
<th>Term/year</th>
<th>Course number</th>
<th>Course title</th>
<th>Semester hrs</th>
<th>Class size</th>
</tr>
</thead>
<tbody>
<tr>
<td>2008S</td>
<td>CS430</td>
<td>Introduction to Algorithms</td>
<td>48</td>
<td>36</td>
</tr>
<tr>
<td>2007F</td>
<td>CS530</td>
<td>Theory of Computing</td>
<td>48</td>
<td>27</td>
</tr>
<tr>
<td>2007S</td>
<td>CS535</td>
<td>Design and Analysis of Algorithms</td>
<td>48</td>
<td>18</td>
</tr>
<tr>
<td>2007S</td>
<td>CS595</td>
<td>Algorithms in Economics and Game Theory</td>
<td>48</td>
<td>3</td>
</tr>
<tr>
<td>2006F</td>
<td>CS533</td>
<td>Computational Geometry</td>
<td>48</td>
<td>5</td>
</tr>
</tbody>
</table>

13. Estimate the percentage of your time devoted to scholarly and/or research activities:
_____%. Please give a brief description of your major research and scholarly
activities:

35%: Research in Algorithms Design, Algorithmic Game Theory, Optimization and
Computational Geometry.

14. If you are a part time faculty member or a full-time faculty member without full-time
commitment to the program, state what percentage of full-time you are assigned to
the program: N/A_____%.
1. Name, current academic rank, and tenure status:

<table>
<thead>
<tr>
<th>Name:</th>
<th>Edward M. Reingold</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rank:</td>
<td>Professor</td>
</tr>
<tr>
<td>Tenure Status:</td>
<td>Tenured</td>
</tr>
</tbody>
</table>

2. Date of original appointment to this faculty, followed by dates and ranks of advancement:

<table>
<thead>
<tr>
<th>Title</th>
<th>Dates Held</th>
</tr>
</thead>
<tbody>
<tr>
<td>Professor of CS, IIT</td>
<td>2000-present</td>
</tr>
<tr>
<td>Emeritus Professor of CS, UIUC</td>
<td>2000-present</td>
</tr>
<tr>
<td>Professor of CS, UIUC</td>
<td>1982-2000</td>
</tr>
<tr>
<td>Associate Professor of CS, UIUC</td>
<td>1974-1982</td>
</tr>
<tr>
<td>Assistant Professor of CS, UIUC</td>
<td>1970-1974</td>
</tr>
</tbody>
</table>

3. Degrees with fields, institutions, and dates

<table>
<thead>
<tr>
<th>Degree</th>
<th>Field</th>
<th>Institution</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>PhD</td>
<td>CS</td>
<td>Cornell University</td>
<td>1971</td>
</tr>
<tr>
<td>MS</td>
<td>CS</td>
<td>Cornell University</td>
<td>1969</td>
</tr>
<tr>
<td>BS</td>
<td>Math</td>
<td>IIT</td>
<td>1967</td>
</tr>
</tbody>
</table>

4. If you do not have an advanced degree in the program area, describe any course work you may have taken, or other ways in which you have achieved competence in the program area; there is no necessity to repeat information here that is contained in later sections of this document.

N/A

5. Conferences, workshops, and professional development programs in which you have participated in the last 5 years to improve teaching and professional competence in the program area:


6. Other related computing experience including teaching, industrial, governmental, etc. (where, when, description and scope of duties):

<table>
<thead>
<tr>
<th>Dates</th>
<th>Where</th>
<th>Duties</th>
</tr>
</thead>
</table>

VI - 21
7. Consulting—list agencies and dates, and briefly describe each project:

<table>
<thead>
<tr>
<th>Dates</th>
<th>Agency</th>
<th>Project</th>
</tr>
</thead>
</table>

8. For the academic year in which the Self Study was written list your assigned duties other than for teaching, (committee membership, advising, etc.) with average hours per week. Indicate which, if any, carry extra compensation. If you are course coordinator for courses taught by other than full-time or part time faculty, please indicate here which courses.

9. For the four years preceding the Self Study, list all department, college, and/or university committees of which you are/were a member including year(s) served:

10. Principal publications during the last five years. Give in standard bibliographic format.


``Average-Case Analysis of Some Plurality Algorithms'' (with L. Alonso), submitted for publication.

``Lower Bounds for Pursuit Problems with Multiple Pursuers'' (with L. Alonso), in preparation.

``Writing Numbers in Words in \TeX,'' TUGboat 28, 2 (2007), 256-259.

11. Other scholarly activity during the last 5 years: grants, sabbaticals, software development, etc.:

12. Courses taught this and last academic year term-by-term. This year is the academic year in which this Self-Study report is prepared; the last year was the year prior to this. If you were on sabbatical leave, please enter the information for the year prior to the sabbatical. Please list each section of the same course separately.

<table>
<thead>
<tr>
<th>Term/year</th>
<th>Course number</th>
<th>Course title</th>
<th>Semester hrs</th>
<th>Class size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fall, 2006</td>
<td>CS 535</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Spring, 2007</td>
<td>CS 330</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Fall, 2007</td>
<td>CS 430</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Spring, 2008</td>
<td>CS 330</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

13. Estimate the percentage of your time devoted to scholarly and/or research activities: 50%. Please give a brief description of your major research and scholarly activities:

14. If you are a part time faculty member or a full-time faculty member without full-time commitment to the program, state what percentage of full-time you are assigned to the program: ______%.
1. Name, current academic rank, and tenure status:

<table>
<thead>
<tr>
<th>Name:</th>
<th>Xian-He Sun</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rank:</td>
<td>Professor</td>
</tr>
<tr>
<td>Tenure Status:</td>
<td>Tenured</td>
</tr>
</tbody>
</table>

2. Date of original appointment to this faculty, followed by dates and ranks of advancement:

<table>
<thead>
<tr>
<th>Title</th>
<th>Dates Held</th>
</tr>
</thead>
<tbody>
<tr>
<td>Professor</td>
<td>Sept. 2002 - Present</td>
</tr>
<tr>
<td>AssociateProfessor</td>
<td>September 1999 – Sept. 2009</td>
</tr>
</tbody>
</table>

3. Degrees with fields, institutions, and dates

<table>
<thead>
<tr>
<th>Degree</th>
<th>Field</th>
<th>Institution</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ph.D</td>
<td>CS</td>
<td>Michigan State University</td>
<td>1990</td>
</tr>
<tr>
<td>MS</td>
<td>CS</td>
<td>MSU</td>
<td>1987</td>
</tr>
<tr>
<td>MS</td>
<td>Math</td>
<td>MSU</td>
<td>1985</td>
</tr>
<tr>
<td>BS</td>
<td>Math</td>
<td>Beijing Normal University, P.R. China</td>
<td>1982</td>
</tr>
</tbody>
</table>

4. If you do not have an advanced degree in the program area, describe any course work you may have taken, or other ways in which you have achieved competence in the program area; there is no necessity to repeat information here that is contained in later sections of this document.

N/A

5. Conferences, workshops, and professional development programs in which you have participated in the last 5 years to improve teaching and professional competence in the program area:

- Symposium: Being Professors in the USA & Promoting International Collaboration, Chicago, April 2005

6. Other related computing experience including teaching, industrial, governmental, etc. (where, when, description and scope of duties):

<table>
<thead>
<tr>
<th>Dates</th>
<th>Where</th>
<th>Duties</th>
</tr>
</thead>
<tbody>
<tr>
<td>9/07-9/07</td>
<td>Fermi National Lab.</td>
<td>Visiting Scientist</td>
</tr>
<tr>
<td>9/99-</td>
<td>Argonne National Lab.</td>
<td>Guest faculty</td>
</tr>
</tbody>
</table>
7. Consulting—list agencies and dates, and briefly describe each project:

<table>
<thead>
<tr>
<th>Dates</th>
<th>Agency</th>
<th>Project</th>
</tr>
</thead>
</table>

8. For the academic year in which the Self Study was written list your assigned duties other than for teaching, (committee membership, advising, etc.) with average hours per week. Indicate which, if any, carry extra compensation. If you are course coordinator for courses taught by other than full-time or part time faculty, please indicate here which courses.

Promotion Committee Chair, 2 hours
Seminar Coordinator, 1 hour

9. For the four years preceding the Self Study, list all department, college, and/or university committees of which you are/were a member including year(s) served:

| Coordinator of departmental Seminar activities (2000-2005, 2007-present) |
| Coordinator of the Best Student Paper Award Committee (1999-2006) |
| Member, Awards Committee (1994-1998, 2000-2006) |
| Member, Chair search committee (2004-2005) |
| Member of IIT Faculty Council (2001-2004) |
| Member of IIT Research Council (2004-2006) |
| Member of IIT Information Technology Advisory Council (2001-present) |
| Member of IIT Technology Committee (2001-present) |
| Member of IIT Research Council Task Force on High Performance Computing (2000-present) |
| Member, Tenure and Promotion Committee, College of Science and Letters, IIT (2005-2006) |

10. Principal publications during the last five years. Give in standard bibliographic format.


14. Xian-He Sun, local PI (IIT), Valerie Taylor, coordinator and local PI (Northwestern University), Ian Foster, local PI (Argonne and University of Chicago), Alok Choudhary, Peter Dinda, Joel Mambretti, Co-PIs, “Collaborative Research: DOT -- Distributed Optical Testbed to Facilitate the Development of Techniques for Efficient Execution of Distributed Applications”, NSF, Collaborative Research, EIA-0224377 Sept. 2002-Aug. 2006.


12. Courses taught this and last academic year term-by-term. This year is the academic year in which this Self-Study report is prepared; the last year was the year prior to this. If you were on sabbatical leave, please enter the information for the year prior to the sabbatical. Please list each section of the same course separately.

<table>
<thead>
<tr>
<th>Term/year</th>
<th>Course number</th>
<th>Course title</th>
<th>Semester hrs</th>
<th>Class size</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Last academic year was on sabbatical</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

13. Estimate the percentage of your time devoted to scholarly and/or research activities: 150%. Please give a brief description of your major research and scholarly activities:

Conduct research in computer science, supervise students, and provide professional services

14. If you are a part time faculty member or a full-time faculty member without full-time commitment to the program, state what percentage of full-time you are assigned to the program: _____%. 
Associate Professor
1. Name, current academic rank, and tenure status:

<table>
<thead>
<tr>
<th>Name:</th>
<th>Gady Agam</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rank:</td>
<td>Associate Professor</td>
</tr>
<tr>
<td>Tenure Status:</td>
<td>tenured</td>
</tr>
</tbody>
</table>

2. Date of original appointment to this faculty, followed by dates and ranks of advancement:

<table>
<thead>
<tr>
<th>Title</th>
<th>Dates Held</th>
</tr>
</thead>
<tbody>
<tr>
<td>Associate professor</td>
<td>2007-present</td>
</tr>
<tr>
<td>Assistant professor</td>
<td>2000-2007</td>
</tr>
</tbody>
</table>

3. Degrees with fields, institutions, and dates

<table>
<thead>
<tr>
<th>Degree</th>
<th>Field</th>
<th>Institution</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>PhD</td>
<td>ECE</td>
<td>Ben-Gurion University</td>
<td>1999</td>
</tr>
<tr>
<td>MSc</td>
<td>ECE</td>
<td>Ben-Gurion University</td>
<td>1994</td>
</tr>
<tr>
<td>BSc</td>
<td>ECE</td>
<td>Ben-Gurion University</td>
<td>1990</td>
</tr>
</tbody>
</table>

4. If you do not have an advanced degree in the program area, describe any course work you may have taken, or other ways in which you have achieved competence in the program area; there is no necessity to repeat information here that is contained in later sections of this document.

1. Conferences, workshops, and professional development programs in which you have participated in the last 5 years to improve teaching and professional competence in the program area:

- IEEE International symposium on biomedical imaging: 2007
- IEEE International conference on image processing: 2006
- IEEE International conference on document image analysis: 2006
- IEEE 3D Digital imaging and modeling: 2005
- Symposium on document image understanding: 2005

6. Other related computing experience including teaching, industrial, governmental, etc. (where, when, description and scope of duties):
7. Consulting—list agencies and dates, and briefly describe each project:

<table>
<thead>
<tr>
<th>Dates</th>
<th>Agency</th>
<th>Project</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

8. For the academic year in which the Self Study was written list your assigned duties other than for teaching, (committee membership, advising, etc.) with average hours per week. Indicate which, if any, carry extra compensation. If you are course coordinator for courses taught by other than full-time or part time faculty, please indicate here which courses.

<table>
<thead>
<tr>
<th>Duties</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS Graduate studies committee chair, 3 hours/week, no extra compensation</td>
</tr>
<tr>
<td>University Graduate studies committee, 1 hours/week, no extra compensation</td>
</tr>
</tbody>
</table>

9. For the four years preceding the Self Study, list all department, college, and/or university committees of which you are/were a member including year(s) served:

<table>
<thead>
<tr>
<th>Committees</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS graduate studies committee: 2001-present</td>
</tr>
<tr>
<td>CS graduate admission: 2000-2007</td>
</tr>
<tr>
<td>Main campus faculty council: 2002-2004</td>
</tr>
<tr>
<td>Main campus academic computing committee: 2002-2004</td>
</tr>
</tbody>
</table>

10. Principal publications during the last five years. Give in standard bibliographic format.


11. Other scholarly activity during the last 5 years: grants, sabbaticals, software development, etc.:
Grants: ARDA Challenge grant $1,295,000 co-PI, NSF-KDD $175,000 co-PI, Pritzker ISFG $40,000 PI


12. Courses taught this and last academic year term-by-term. This year is the academic year in which this Self-Study report is prepared; the last year was the year prior to this. If you were on sabbatical leave, please enter the information for the year prior to the sabbatical. Please list each section of the same course separately.

<table>
<thead>
<tr>
<th>Term/year</th>
<th>Course number</th>
<th>Course title</th>
<th>Semester hrs</th>
<th>Class size</th>
</tr>
</thead>
<tbody>
<tr>
<td>F07</td>
<td>CS100</td>
<td>Introduction to professions</td>
<td>3</td>
<td>25</td>
</tr>
<tr>
<td></td>
<td>CS411</td>
<td>Computer graphics</td>
<td>3</td>
<td>30</td>
</tr>
<tr>
<td>S07</td>
<td>CS512</td>
<td>Computer vision</td>
<td>3</td>
<td>10</td>
</tr>
<tr>
<td>F06</td>
<td>CS511</td>
<td>Topics in computer graphics</td>
<td>3</td>
<td>30</td>
</tr>
<tr>
<td>S06</td>
<td>CS512</td>
<td>Computer Vision</td>
<td>3</td>
<td>20</td>
</tr>
</tbody>
</table>

13. Estimate the percentage of your time devoted to scholarly and/or research activities: 60%. Please give a brief description of your major research and scholarly activities:

General areas: computer vision, computer graphics, pattern recognition, machine learning
Specific applications: 3D medical imaging (CT & DT-MRI), document imaging, geometric modeling

14. If you are a part time faculty member or a full-time faculty member without full-time commitment to the program, state what percentage of full-time you are assigned to the program: _____%.
1. Name, current academic rank, and tenure status:

<table>
<thead>
<tr>
<th>Name:</th>
<th>Shlomo Argamon</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rank:</td>
<td>Associate Professor</td>
</tr>
<tr>
<td>Tenure Status:</td>
<td>Tenured</td>
</tr>
</tbody>
</table>

2. Date of original appointment to this faculty, followed by dates and ranks of advancement:

<table>
<thead>
<tr>
<th>Title</th>
<th>Dates Held</th>
</tr>
</thead>
<tbody>
<tr>
<td>Associate Professor (untenured)</td>
<td>8/2002 – 5/2006</td>
</tr>
<tr>
<td>Associate Professor (tenured)</td>
<td>5/2006 – present</td>
</tr>
</tbody>
</table>

3. Degrees with fields, institutions, and dates

<table>
<thead>
<tr>
<th>Degree</th>
<th>Field</th>
<th>Institution</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>M.Phil</td>
<td>Computer Science</td>
<td>Yale University</td>
<td>12/91</td>
</tr>
<tr>
<td>Ph.D.</td>
<td>Computer Science</td>
<td>Yale University</td>
<td>5/94</td>
</tr>
</tbody>
</table>

4. If you do not have an advanced degree in the program area, describe any course work you may have taken, or other ways in which you have achieved competence in the program area; there is no necessity to repeat information here that is contained in later sections of this document.

5. Conferences, workshops, and professional development programs in which you have participated in the last 5 years to improve teaching and professional competence in the program area:

- Chicago Colloquium on Digital Humanities and Computer Science
- ACM Conference on Information and Knowledge Management
- Conference of the Association for Computational Linguistics (ACL)
- Association for Artificial Intelligence
- ACM SIGIR Conference on Research & Development on Information Retrieval
6. Other related computing experience including teaching, industrial, governmental, etc.
(where, when, description and scope of duties):

<table>
<thead>
<tr>
<th>Dates</th>
<th>Where</th>
<th>Duties</th>
</tr>
</thead>
<tbody>
<tr>
<td>1999-2002</td>
<td>Jerusalem College of Technology</td>
<td>Assistant Professor; teaching undergraduates and advising student projects</td>
</tr>
<tr>
<td>1994-2002</td>
<td>Bar-Ilan University</td>
<td>Research fellow and affiliate; research in computer science and advising master’s theses and projects</td>
</tr>
<tr>
<td>6-7/1998</td>
<td>Rutgers University</td>
<td>Visiting research scientist; research on computational design</td>
</tr>
<tr>
<td>5-8/1994</td>
<td>University of Chicago</td>
<td>Visiting postdoctoral fellow; research on mobile robotics</td>
</tr>
</tbody>
</table>

7. Consulting—list agencies and dates, and briefly describe each project:

<table>
<thead>
<tr>
<th>Dates</th>
<th>Agency</th>
<th>Project</th>
</tr>
</thead>
</table>

8. For the academic year in which the Self Study was written list your assigned duties other than for teaching, (committee membership, advising, etc.) with average hours per week. Indicate which, if any, carry extra compensation. If you are course coordinator for courses taught by other than full-time or part time faculty, please indicate here which courses.

| Department Undergraduate Committee: 2 hrs/week |
| University Faculty Council: 2 hrs/week |
| Advising students: 15 hrs/week |

9. For the four years preceding the Self Study, list all department, college, and/or university committees of which you are/were a member including year(s) served:

| Dept. Undergraduate Committee: 2005 – present |
10. Principal publications during the last five years. Give in standard bibliographic format.

Shlomo Argamon and Moshe Koppel. Automatically profiling the author of an anonymous text. Communications of the Association for Computing Machinery, in press.


11. Other scholarly activity during the last 5 years: grants, sabbaticals, software development, etc.:

<table>
<thead>
<tr>
<th>Grant Description</th>
<th>Amount</th>
<th>Dates</th>
<th>PI(s)</th>
<th>Collaborators</th>
</tr>
</thead>
<tbody>
<tr>
<td>NSF CISE/EI: An Undergraduate CS Specialization in Knowledge Management Systems;</td>
<td>$269,999</td>
<td>10/1/04 to 9/30/08</td>
<td>Shlomo Argamon (IIT)</td>
<td>Nazli Goharian (IIT), Ophir Frieder (IIT), David Grossman (IIT), Nambury Raju (IIT).</td>
</tr>
<tr>
<td>NSF CISE/EI: An Undergraduate CS Specialization in Knowledge Management Systems:</td>
<td>$12,000</td>
<td>REU Supplement</td>
<td>Shlomo Argamon (IIT)</td>
<td>Nazli Goharian (IIT), Ophir Frieder (IIT), David Grossman (IIT), Nambury Raju (IIT).</td>
</tr>
<tr>
<td>Binational Science Foundation (BSF): Collaborative Proposal: Automated Text Analytic Methods to Assess Psychological Attributes of Authors (joint with Bar-Ilan University and University of Texas at Austin);</td>
<td>$90,000</td>
<td>9/1/04 through 2/29/08</td>
<td>Shlomo Argamon (IIT)</td>
<td>Moshe Koppel (Bar-Ilan), James Pennebaker (U. Texas).</td>
</tr>
<tr>
<td>ARDA Challenge Workshop: Complex Document Information Processing (CDIP);</td>
<td>$1,069,000</td>
<td>1/1/05 to 12/31/06</td>
<td>Ophir Frieder (IIT)</td>
<td>Gady Agam (IIT), Shlomo Argamon (IIT), Nazli Goharian (IIT), David Grossman (IIT), David D. Lewis (consultant).</td>
</tr>
<tr>
<td>NSF CISE/CRI: Community Resources for Research in Automated Authorship Attribution (joint with Rutgers U.);</td>
<td>$43,476</td>
<td>7/15/05 through 6/30/06</td>
<td>Shlomo Argamon</td>
<td>David Madigan (Rutgers), consultant David D. Lewis.</td>
</tr>
<tr>
<td>BM/EAS KDD Challenge: Meta-Learning and Computational Stylistics for Authorship Entity Recognition;</td>
<td>$61,575</td>
<td>7/15/05 to 9/30/05</td>
<td>Shlomo Argamon</td>
<td>Ophir Frieder (IIT), David Grossman (IIT).</td>
</tr>
</tbody>
</table>

12. Courses taught this and last academic year term-by-term. This year is the academic year in which this Self-Study report is prepared; the last year was the year prior to this. If you were on sabbatical leave, please enter the information for the year prior to the sabbatical. Please list each section of the same course separately.

<table>
<thead>
<tr>
<th>Term/year</th>
<th>Course number</th>
<th>Course title</th>
<th>Semester hours</th>
<th>Class size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fall 2006</td>
<td>CS 481</td>
<td>Artificial Intelligence: Natural Language Understanding</td>
<td>3</td>
<td>15</td>
</tr>
<tr>
<td>Spring 2007</td>
<td>CS 480</td>
<td>Artificial Intelligence: Planning and Control</td>
<td>3</td>
<td>30</td>
</tr>
<tr>
<td>Spring 2007</td>
<td>CS 584</td>
<td>Machine Learning (graduate)</td>
<td>3</td>
<td>10</td>
</tr>
<tr>
<td>Fall 2007</td>
<td>CS 100</td>
<td>Introduction to the Profession</td>
<td>3</td>
<td>30</td>
</tr>
<tr>
<td>Fall 2007</td>
<td>CS 481</td>
<td>Artificial Intelligence: Natural Language Understanding</td>
<td>3</td>
<td>10</td>
</tr>
<tr>
<td>Spring 2007</td>
<td>CS 480</td>
<td>Artificial Intelligence: Planning and Control</td>
<td>3</td>
<td>30</td>
</tr>
<tr>
<td>Spring 2007</td>
<td>CS 482</td>
<td>Information and Knowledge Management Systems</td>
<td>3</td>
<td>5</td>
</tr>
</tbody>
</table>
13. Estimate the percentage of your time devoted to scholarly and/or research activities: 60%. Please give a brief description of your major research and scholarly activities:

Development and application of machine learning methods for computational analysis of human language semantics and style. Particularly in the areas of sentiment analysis, stylistics, and authorship, with applications to forensics and humanities scholarship.

14. If you are a part time faculty member or a full-time faculty member without full-time commitment to the program, state what percentage of full-time you are assigned to the program: ____%.
1. Name, current academic rank, and tenure status:
Name: Gruia Calinescu
Rank: Associate Professor
Tenure Status: tenured

2. Date of original appointment to this faculty, followed by dates and ranks of advancement:
February 2000 - Assistant Professor
September 2007 - Associate Professor
Research Associate, May-August 2001
Postdoctoral Fellow September 1998-August 1999

3. Degrees with fields, institutions, and dates
Degree PhD
Field Algorithms, Combinatorics and Optimization
Institution Georgia Institute of Technology
Data September 1998

4. If you do not have an advanced degree in the program area, describe any course work you may have taken, or other ways in which you have achieved competence in the program area; there is no necessity to repeat information here that is contained in later sections of this document.

5. Conferences, workshops, and professional development programs in which you have participated in the last 5 years to improve teaching and professional competence in the program area:

6. Other related computing experience including teaching, industrial, governmental, etc. (where, when, description and scope of duties):
NSF panelist
Dates March 2003 and May 2007
Where Washington, DC
Duties review research proposals

7. Consulting-list agencies and dates, and briefly describe each project:

8. For the academic year in which the Self Study was written list your assigned duties other than for teaching, (committee membership, advising, etc.) with average hours per week. Indicate which, if any, carry extra compensation. If you are course coordinator for courses taught by other than full-time or part time faculty, please indicate here which courses.
PhD Coordinator, 3h/week, Graduate Committee 1h/week, Graduate admission 1h/week, Advising 1h/week (none has extra compensation, and the advising of my PhD student is not included above)

9. For the four years preceding the Self Study, list all department, college, and/or university committees of which you are/were a member including year(s) served: graduate committee 2002-present, graduate admission committee - 2000-present, Best Student Paper Award Committee, 2003

10. Principal publications during the last five years. Give in standard bibliographic format.

``Maximizing a Submodular Set Function subject to a Matroid Constraint,'' by G. Calinescu, C. Chekuri, M. Pal and J. Vondrak, accepted at the 12th Conference on Integer Programming and Combinatorial Optimization (IPCO 2007).

11. Other scholarly activity during the last 5 years: grants, sabbaticals, software development, etc.:

12. Courses taught this and last academic year term-by-term. This year is the academic year in which this Self-Study report is prepared; the last year was the year prior to this. If you were on sabbatical leave, please enter the information for the year prior to the sabbatical. Please list each section of the same course separately.
   Fall 2007 CS 330-001 Discrete Structures 3 credits 20 students
   Fall 2007 CS 397-001 Programming Competition 1 credit 4 students
   Spring 2007 CS 538-001 Combinatorial Optimization 3 credits 10 students
   Spring 2007 CS 538-251 Combinatorial Optimization 3 credits 6 students
   Spring 2007 CS 695-001 Doctoral Seminar 1 credit 10 students
   Spring 2007 CS 695-092 Doctoral Seminar 1 credit 1 student
   Fall 2006 CS 530-001 Theory of Computation 3 credits 19 students
   Fall 2006 CS 530-001 Theory of Computation 3 credits 19 students
   Fall 2006 CS 530-091 Theory of Computation 3 credits 3 students
Fall 2006 CS 530-251 Theory of Computation 3 credits 5 students

13. Estimate the percentage of your time devoted to scholarly and/or research activities: 45__%. Please give a brief description of your major research and scholarly activities:
Approximation Algorithms: provide provably fast, approximate solutions to NP-Hard optimization problems.

14. If you are a part time faculty member or a full-time faculty member without full-time commitment to the program, state what percentage of full-time you are assigned to the program: _____%.
1. Name, current academic rank, and tenure status:

<table>
<thead>
<tr>
<th>Name</th>
<th>Tzilla Elrad</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rank</td>
<td>Research Professor</td>
</tr>
<tr>
<td>Tenure Status</td>
<td>none</td>
</tr>
</tbody>
</table>

2. Date of original appointment to this faculty, followed by dates and ranks of advancement:

<table>
<thead>
<tr>
<th>Title</th>
<th>Dates Held</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assistant Prof.</td>
<td>- 1982</td>
</tr>
<tr>
<td>Associate Prof.</td>
<td>- 1988</td>
</tr>
<tr>
<td>Research Prof.</td>
<td>- 1995</td>
</tr>
</tbody>
</table>

3. Degrees with fields, institutions, and dates

<table>
<thead>
<tr>
<th>Degree</th>
<th>Field</th>
<th>Institution</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bs</td>
<td>Mathematic</td>
<td>Hebrew U</td>
<td>1969</td>
</tr>
<tr>
<td>BS</td>
<td>Physics</td>
<td>Hebrew U</td>
<td>1969</td>
</tr>
<tr>
<td>MS</td>
<td>Computer Science</td>
<td>Syracuse U</td>
<td>1978</td>
</tr>
<tr>
<td>PHD</td>
<td>Computer Science</td>
<td>Thechnition</td>
<td>1982</td>
</tr>
</tbody>
</table>

Publications for 2007


Mahoney, M., Elrad, T. A Pattern Story for Aspect-Oriented State Machines, 14 Conference On Pattern Languages of Programs, September 2007


http://www.iit.edu/~concur/weavr/papers/stateful_AOM_paper.pdf

Cottenier, T., van den Berg, A., Elrad, T. Joinpoint Inference from Behavioral Specification to Implementation, ECOOP07

http://www.iit.edu/~concur/weavr/papers/ecoop_state_paper.pdf

Cottenier, T. Motorola WEAVR: Aspect-Oriented Modeling for Simulation and Code Generation. Tutorial at ECOOP'07


Publications for -2006

http://www.iit.edu/~concur/weavr/papers/stateful_AOM_paper.pdf

Cottenier, T., van den Berg, A., Elrad, T. Joinpoint Inference from Behavioral Specification to Implementation, to be presented at ECOOP'07, 2007
http://www.iit.edu/~concur/weavr/papers/ecoop_state_paper.pdf


Cottenier, T. Aspect-Oriented Modeling and Simulation. Tutorial Proposal to be presented at AOSD'07, 2006
http://www.iit.edu/~concur/weavr/papers/AOMS_tutorial_proposal.pdf

Cottenier, T., van den Berg, A., Elrad, T. The Motorola WEAVR: Model Weaving in a Large Industrial Context. Industry Track paper to be presented at AOSD'07, 2006
http://www.iit.edu/~concur/weavr/papers/AOSD_MotorolaWEAVR.pdf

http://www.iit.edu/~concur/weavr/papers/motorola_weavr.pdf

http://www.ppci.ca/?q=bssn

http://www.iit.edu/~concur/weavr/papers/motorola_weavr.pdf

Prunicki, A., Elrad, T. ACLamate: An AOSD Security Framework for Access Control. The 2nd IEEE International Symposium on Dependable, Autonomic and Secure Computing (DASC'06) at Indiana University, Purdue University, Indianapolis, USA, September 29-October 1, 2006
http://doi.ieeecomputersociety.org/10.1109/DASC.2006.16

http://www.aspect-modeling.org/acceptedpapers_oct06/thomas.pdf

Cottenier, T., van den Berg, A., Elrad, T. Model Weaving: Bridging the Divide between Translationists and Elaborationists. Workshop on Aspect-Oriented Modeling at the 9th International

VI - 46
Conference on Model Driven Engineering Languages and Systems, Milan, Italy, 2006
http://www.aspect-modeling.org/acceptedpapers_oct06/thomas.pdf

http://www.cis.uab.edu/gray/Pubs/aom-models-2006.pdf

http://people.cs.uchicago.edu/~borja/gw06/devtools/

http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?isnumber=34697&arnumber=1656059&count=475&index=347

http://www.csupomona.edu/~wtsi/index.htm

Sufyan Almajali, Tzilla Elrad, Benefits and Challenges of a Class-Based Design for Dynamic Aspects in DAC++, Proceedings of the 2006 Software Engineering Properties of Languages and Aspect Technologies Workshop (SPLAT'06) as part of AOSD'06 (Bonn, Germany, March 2006)
http://www.aosd.net/workshops/splat/2006/

Sufyan Almajali, Tzilla Elrad, An Open Aspect Oriented Software Design with an Interface to a Third-Party Software, Proceedings of the 2006 Open Aspect Languages Workshop (OAL'06) as part of AOSD'06 (Bonn, Germany, March 2006)
http://www.aosd.net/workshops/oal/2006/

http://www.iaria.org/conferences/ICN06.html

Sufyan Almajali, Tzilla Elrad, NDAC++ for Network Aspects, Demonstration as part of the 5th International Conference on Aspect-Oriented Software Development (AOSD'06) (Bonn, Germany, March 2006)
http://www.aosd.net/2006/demos/ndac.php

Publications for 2004-2005

http://www.iaria.org/conferences/ICN06.html


Thomas Cottenier, Aswin Van Den Berg, Tzilla Elrad, **Modeling Aspect-Oriented Compositions**, in proceedings of the 7th International Workshop on Aspect-Oriented Modeling, in conjunction of the 8th International Conference on Model Driven Engineering Languages and Systems (MoDELS'05) (formerly the UML series of conferences), Montego Bay, Jamaica, October 2005 http://dawis.informatik.uni-essen.de/events/AOM_MODELS2005/

Thomas Cottenier, Aswin Van Den Berg, Tzilla Elrad, **Modeling Aspect-Oriented Compositions**, MoDELS'05 satellite proceedings http://www.springer.com sgw/cda/fronpage/0,11855,5-40109-22-121240522-0,00.html

Mark Mahoney, Tzilla Elrad, **Modeling Platform Specific Attributes of a System as Crosscutting Concerns using Aspect-Oriented Statecharts and Virtual Finite State Machines**, the 6th International Workshop on Aspect-Oriented Modeling in Conjunction with AOSD ’05 International Conference on Aspect Oriented Software Development, Chicago, USA, March 2005 http://dawis.informatik.uni-essen.de/events/AOM_AOSD2005/Mahoney.pdf


Cottenier, T., Elrad, T., Linguistic provisions for Aspect/Core semantic interactions, Workshop on Software Engineering Properties of Languages for Aspect Technologies as
En-Hsin Huang and Tzilla Elrad, **Intelligent Resource Agents For Embedded Systems**
*ESA-04* - The 2004 International Conference on Embedded Systems and Applications
June 2004
http://juliet.stfx.ca/~lyang/esa-04/


http://dares.enst-bretagne.fr/dares2004/

12. Courses taught this and last academic year term-by-term. This year is the academic year in which this Self-Study report is prepared; the last year was the year prior to this. If you were on sabbatical leave, please enter the information for the year prior to the sabbatical. Please list each section of the same course separately.

<table>
<thead>
<tr>
<th>Term/year</th>
<th>Course number</th>
<th>Course title</th>
<th>Semester hrs</th>
<th>Class size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spring 08</td>
<td>CS536</td>
<td>Science of Programming</td>
<td>45</td>
<td>~70</td>
</tr>
<tr>
<td>Fall 08</td>
<td>CS545</td>
<td>Concurrent Programming</td>
<td>45</td>
<td>~20</td>
</tr>
<tr>
<td>Spring 07</td>
<td>CS536</td>
<td>Science of Programming</td>
<td>45</td>
<td>~70</td>
</tr>
<tr>
<td>Fall 07</td>
<td>CS545</td>
<td>Concurrent Programming</td>
<td>45</td>
<td>~20</td>
</tr>
<tr>
<td>Spring 06</td>
<td>CS536</td>
<td>Science of Programming</td>
<td>45</td>
<td>~70</td>
</tr>
<tr>
<td>Fall 0</td>
<td>CS545</td>
<td>Concurrent Programming</td>
<td>45</td>
<td>~20</td>
</tr>
<tr>
<td>Spring 05</td>
<td>CS536</td>
<td>Science of Programming</td>
<td>45</td>
<td>~70</td>
</tr>
<tr>
<td>Fall 0</td>
<td>CS545</td>
<td>Concurrent Programming</td>
<td>45</td>
<td>~20</td>
</tr>
<tr>
<td>Spring 04</td>
<td>CS536</td>
<td>Science of Programming</td>
<td>45</td>
<td>~70</td>
</tr>
<tr>
<td>Fall 0</td>
<td>CS545</td>
<td>Concurrent Programming</td>
<td>45</td>
<td>~20</td>
</tr>
<tr>
<td>Spring 03</td>
<td>CS536</td>
<td>Science of Programming</td>
<td>45</td>
<td>~70</td>
</tr>
</tbody>
</table>

13. Estimate the percentage of your time devoted to scholarly and/or research activities: 
______%. Please give a brief description of your major research and scholarly activities:
14. If you are a part time faculty member or a full-time faculty member without full-time commitment to the program, state what percentage of full-time you are assigned to the program: _____%.
1. Name, current academic rank, and tenure status:

<table>
<thead>
<tr>
<th>Name:</th>
<th>David Grossman</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rank:</td>
<td>Associate Professor</td>
</tr>
<tr>
<td>Tenure Status:</td>
<td>Tenured</td>
</tr>
</tbody>
</table>

2. Date of original appointment to this faculty, followed by dates and ranks of advancement:

<table>
<thead>
<tr>
<th>Title</th>
<th>Dates Held</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assistant Professor</td>
<td>Fall 1999 – Spring 2005</td>
</tr>
<tr>
<td>Associate Professor</td>
<td>Fall 2006 - present</td>
</tr>
</tbody>
</table>

3. Degrees with fields, institutions, and dates

<table>
<thead>
<tr>
<th>Degree</th>
<th>Field</th>
<th>Institution</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>BS</td>
<td>CS</td>
<td>Clemson University</td>
<td>1985</td>
</tr>
<tr>
<td>MS</td>
<td>CS</td>
<td>American University</td>
<td>1988</td>
</tr>
<tr>
<td>PhD</td>
<td>Information</td>
<td>George Mason University</td>
<td>1995</td>
</tr>
<tr>
<td></td>
<td>Technology</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

4. If you do not have an advanced degree in the program area, describe any course work you may have taken, or other ways in which you have achieved competence in the program area; there is no necessity to repeat information here that is contained in later sections of this document.

N/A

5. Conferences, workshops, and professional development programs in which you have participated in the last 5 years to improve teaching and professional competence in the program area:

<table>
<thead>
<tr>
<th>Program</th>
<th>Years</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACM SIGIR</td>
<td>multiple years</td>
</tr>
<tr>
<td>ACM CIKM</td>
<td>multiple years, General Chair 2002</td>
</tr>
<tr>
<td>NIST TREC</td>
<td>multiple years</td>
</tr>
</tbody>
</table>

6. Other related computing experience including teaching, industrial, governmental, etc. (where, when, description and scope of duties):

<table>
<thead>
<tr>
<th>Dates</th>
<th>Where</th>
<th>Duties</th>
</tr>
</thead>
</table>
7. Consulting—list agencies and dates, and briefly describe each project:

<table>
<thead>
<tr>
<th>Dates</th>
<th>Agency</th>
<th>Project</th>
</tr>
</thead>
<tbody>
<tr>
<td>1999-2001</td>
<td>IITRI</td>
<td>Data Architecture</td>
</tr>
<tr>
<td>2001-2002</td>
<td>BAE</td>
<td>Data Mining for large Inf. System</td>
</tr>
<tr>
<td>2004-present</td>
<td>SAIC</td>
<td>A variety of large search systems</td>
</tr>
</tbody>
</table>

8. For the academic year in which the Self Study was written list your assigned duties other than for teaching, (committee membership, advising, etc.) with average hours per week. Indicate which, if any, carry extra compensation. If you are course coordinator for courses taught by other than full-time or part time faculty, please indicate here which courses.

- Chair, Undergraduate Studies Committee, 3-5 hours per week
- Member, University IPRO Committee, 2-4 hours per week

9. For the four years preceding the Self Study, list all department, college, and/or university committees of which you are/were a member including year(s) served:

- CS Tenure & Promotion – all years
- University Wide Tenure & Promotion – 2003 – 2005
- Chairman Review Committee – 2004

10. Principal publications during the last five years. Give in standard bibliographic format.

11. Other scholarly activity during the last 5 years: grants, sabbaticals, software development, etc.:


12. Courses taught this and last academic year term-by-term. This year is the academic year in which this Self-Study report is prepared; the last year was the year prior to this. If you were on sabbatical leave, please enter the information for the year prior to the sabbatical. Please list each section of the same course separately.

<table>
<thead>
<tr>
<th>Term/year</th>
<th>Course number</th>
<th>Course title</th>
<th>Semester hours</th>
<th>Class size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fall 07</td>
<td>CS522</td>
<td>Advanced Data Mining</td>
<td>3</td>
<td>7</td>
</tr>
<tr>
<td>Spring 08</td>
<td>CS529</td>
<td>Advanced Information Retrieval</td>
<td>3</td>
<td>3</td>
</tr>
</tbody>
</table>

13. Estimate the percentage of your time devoted to scholarly and/or research activities: **__70___%**. Please give a brief description of your major research and scholarly activities:

Scalable search solutions for text, XML, and structured data. Associate director of the IIT Information Retrieval Lab (www.ir.iit.edu).

14. If you are a part time faculty member or a full-time faculty member without full-time commitment to the program, state what percentage of full-time you are assigned to the program: **N/A_____%**.
1. Name, current academic rank, and tenure status:

<table>
<thead>
<tr>
<th>Name</th>
<th>Cindy Hood</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rank</td>
<td>Associate Professor</td>
</tr>
<tr>
<td>Tenure Status</td>
<td>Tenured</td>
</tr>
</tbody>
</table>

2. Date of original appointment to this faculty, followed by dates and ranks of advancement:

<table>
<thead>
<tr>
<th>Title</th>
<th>Dates Held</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assistant Professor</td>
<td>8/96-8/04</td>
</tr>
<tr>
<td>Associate Professor</td>
<td>8/04-present</td>
</tr>
</tbody>
</table>

3. Degrees with fields, institutions, and dates

<table>
<thead>
<tr>
<th>Degree</th>
<th>Field</th>
<th>Institution</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>BS</td>
<td>Computer and Systems Engineering</td>
<td>Rensselaer Polytechnic Institute</td>
<td>1987</td>
</tr>
<tr>
<td>ME</td>
<td>Electrical Engineering</td>
<td>Stevens Institute of Technology</td>
<td>1990</td>
</tr>
<tr>
<td>PhD</td>
<td>Computer and Systems Engineering</td>
<td>Rensselaer Polytechnic Institute</td>
<td>1997</td>
</tr>
</tbody>
</table>

4. If you do not have an advanced degree in the program area, describe any course work you may have taken, or other ways in which you have achieved competence in the program area; there is no necessity to repeat information here that is contained in later sections of this document.

5. Conferences, workshops, and professional development programs in which you have participated in the last 5 years to improve teaching and professional competence in the program area:

NOMS, IM, ITiCSE, ARES, CCNC, CRN, DYSPAN, Globecom, DSOM, IPOM, ICC
6. Other related computing experience including teaching, industrial, governmental, etc. (where, when, description and scope of duties):

<table>
<thead>
<tr>
<th>Dates</th>
<th>Where</th>
<th>Duties</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

7. Consulting—list agencies and dates, and briefly describe each project:

<table>
<thead>
<tr>
<th>Dates</th>
<th>Agency</th>
<th>Project</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

8. For the academic year in which the Self Study was written list your assigned duties other than for teaching, (committee membership, advising, etc.) with average hours per week. Indicate which, if any, carry extra compensation. If you are course coordinator for courses taught by other than full-time or part time faculty, please indicate here which courses.

<table>
<thead>
<tr>
<th>Duties</th>
</tr>
</thead>
<tbody>
<tr>
<td>Associate Chair – 20 hrs/week (extra compensation)</td>
</tr>
<tr>
<td>Dean Search Committee – 5hrs/week</td>
</tr>
<tr>
<td>Chair of Graduate Admission Committee – 2hrs/week</td>
</tr>
<tr>
<td>Graduate Committee – 1hr/week</td>
</tr>
<tr>
<td>Graduate Student Advising – 2hrs/week</td>
</tr>
</tbody>
</table>

9. For the four years preceding the Self Study, list all department, college, and/or university committees of which you are/were a member including year(s) served:

<table>
<thead>
<tr>
<th>Committees</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dean Search Committee (2007-2008), University Graduate Studies Committee (2004-2006)</td>
</tr>
<tr>
<td>Department Graduate Admissions (2005-present), Chair (2006-present)</td>
</tr>
<tr>
<td>Department Tenure and Promotion Committee (2004-present)</td>
</tr>
<tr>
<td>Department Graduate Committee (2002-present)</td>
</tr>
<tr>
<td>Department Chair Search Committee (2004-present)</td>
</tr>
</tbody>
</table>

10. Principal publications during the last five years. Give in standard bibliographic format.

VI - 56


11. Other scholarly activity during the last 5 years: grants, sabbaticals, software development, etc.: 

12. Courses taught this and last academic year term-by-term. This year is the academic year in which this Self-Study report is prepared; the last year was the year prior to this. If you were on sabbatical leave, please enter the information for the year prior to the sabbatical. Please list each section of the same course separately.

<table>
<thead>
<tr>
<th>Term/year</th>
<th>Course number</th>
<th>Course title</th>
<th>Semester hrs</th>
<th>Class size</th>
</tr>
</thead>
<tbody>
<tr>
<td>S 2008</td>
<td>CS 350</td>
<td>Computer Organization and Assembly Language</td>
<td>3</td>
<td>35</td>
</tr>
<tr>
<td>S 2008</td>
<td>CS 595</td>
<td>Advanced Wireless Network Performance</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>F 2007</td>
<td>CS 350</td>
<td>Computer Organization and Assembly Language</td>
<td>3</td>
<td>33</td>
</tr>
<tr>
<td>S 2007</td>
<td>CS 350</td>
<td>Computer Organization and Assembly Language</td>
<td>3</td>
<td>24</td>
</tr>
<tr>
<td>S 2007</td>
<td>IPRO 305</td>
<td>Building a Wireless Broadband Infrastructure to Support Maritime Applications</td>
<td>3</td>
<td>9</td>
</tr>
</tbody>
</table>

13. Estimate the percentage of your time devoted to scholarly and/or research activities: 
   _____%. Please give a brief description of your major research and scholarly activities:

14. If you are a part time faculty member or a full-time faculty member without full-time commitment to the program, state what percentage of full-time you are assigned to the program: _____%.
1. Name, current academic rank, and tenure status:

<table>
<thead>
<tr>
<th>Name:</th>
<th>Bogdan Korel</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rank:</td>
<td>Associate Professor</td>
</tr>
<tr>
<td>Tenure Status:</td>
<td>Tenured</td>
</tr>
</tbody>
</table>

2. Date of original appointment to this faculty, followed by dates and ranks of advancement:

<table>
<thead>
<tr>
<th>Title</th>
<th>Dates Held</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assistant Professor</td>
<td>1994-present</td>
</tr>
<tr>
<td>Associate Professor</td>
<td>1986-1994</td>
</tr>
</tbody>
</table>

3. Degrees with fields, institutions, and dates

<table>
<thead>
<tr>
<th>Degree</th>
<th>Field</th>
<th>Institution</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>PhD</td>
<td>CS</td>
<td>Oakland University</td>
<td>1986</td>
</tr>
<tr>
<td>MS</td>
<td>Control Systems</td>
<td>Technical University of Kiev</td>
<td></td>
</tr>
</tbody>
</table>

4. If you do not have an advanced degree in the program area, describe any coursework you may have taken, or other ways in which you have achieved competence in the program area; there is no necessity to repeat information here that is contained in later sections of this document.

5. Conferences, workshops, and professional development programs in which you have participated in the last 5 years to improve teaching and professional competence in the program area:

- IEEE International Conference on Software Maintenance
- IEEE International Workshop on Source Code Analysis and Manipulation
- ACM International Symposium on Software Testing and Analysis
- IEEE International Workshop on Program Comprehension
- ACM SigSoft International Symposium on Software Testing and Analysis

6. Other related computing experience including teaching, industrial, governmental, etc. (where, when, description and scope of duties):

<table>
<thead>
<tr>
<th>Dates</th>
<th>Where</th>
<th>Duties</th>
</tr>
</thead>
</table>

VI - 61
7. Consulting—list agencies and dates, and briefly describe each project:

<table>
<thead>
<tr>
<th>Dates</th>
<th>Agency</th>
<th>Project</th>
</tr>
</thead>
</table>

8. For the academic year in which the Self Study was written list your assigned duties other than for teaching, (committee membership, advising, etc.) with average hours per week. Indicate which, if any, carry extra compensation. If you are course coordinator for courses taught by other than full-time or part time faculty, please indicate here which courses.

Interim Chair
Graduate studies committee

9. For the four years preceding the Self Study, list all department, college, and/or university committees of which you are/were a member including year(s) served:

<table>
<thead>
<tr>
<th>Committee Name</th>
<th>Year(s) Served</th>
</tr>
</thead>
<tbody>
<tr>
<td>Graduate studies committee</td>
<td>2004-present</td>
</tr>
<tr>
<td>University Faculty Council</td>
<td>2004-2006</td>
</tr>
<tr>
<td>University Sabbatical Leave Committee</td>
<td>2005</td>
</tr>
<tr>
<td>CS Chair Search Committee</td>
<td>2005-2006</td>
</tr>
<tr>
<td>Tenure and Promotion committee</td>
<td>2004-present</td>
</tr>
</tbody>
</table>

10. Principal publications during the last five years. Give in standard bibliographic format.


11. Other scholarly activity during the last 5 years: grants, sabbaticals, software development, etc.:

12. Courses taught this and last academic year term-by-term. This year is the academic year in which this Self-Study report is prepared; the last year was the year prior to this. If you were on sabbatical leave, please enter the information for the year prior to the sabbatical. Please list each section of the same course separately.

<table>
<thead>
<tr>
<th>Term/year</th>
<th>Course number</th>
<th>Course title</th>
<th>Semester hrs</th>
<th>Class size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spring 08</td>
<td>CS586</td>
<td>Software Architecture</td>
<td>3</td>
<td>76</td>
</tr>
<tr>
<td>Fall 07</td>
<td>CS599</td>
<td>Software Testing and Analysis</td>
<td>3</td>
<td>73</td>
</tr>
<tr>
<td>Spring 07</td>
<td>CS 590</td>
<td>Seminar in Advanced Software Analysis</td>
<td>3</td>
<td>12</td>
</tr>
<tr>
<td>Fall 06</td>
<td>CS 537</td>
<td>Software Metrics</td>
<td>3</td>
<td>66</td>
</tr>
</tbody>
</table>

13. Estimate the percentage of your time devoted to scholarly and/or research activities: 25%. Please give a brief description of your major research and scholarly activities:

Software engineering: automated software testing and analysis, program slicing, software maintenance and comprehension.

14. If you are a part time faculty member or a full-time faculty member without full-time commitment to the program, state what percentage of full-time you are assigned to the program: ____%.
1. Name, current academic rank, and tenure status:

<table>
<thead>
<tr>
<th>Name:</th>
<th>Xiang-Yang Li</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rank:</td>
<td>Associate Professor</td>
</tr>
<tr>
<td>Tenure Status:</td>
<td>Tenured</td>
</tr>
</tbody>
</table>

2. Date of original appointment to this faculty, followed by dates and ranks of advancement:

<table>
<thead>
<tr>
<th>Title</th>
<th>Dates Held</th>
</tr>
</thead>
<tbody>
<tr>
<td>Associate Professor of CS, IIT</td>
<td>2006-present</td>
</tr>
<tr>
<td>Assistant Professor of CS, IIT</td>
<td>2000-2006</td>
</tr>
</tbody>
</table>

3. Degrees with fields, institutions, and dates

<table>
<thead>
<tr>
<th>Degree</th>
<th>Field</th>
<th>Institution</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>PhD</td>
<td>CS</td>
<td>University of Illinois at Urbana Champaign</td>
<td>2000</td>
</tr>
<tr>
<td>MS</td>
<td>CS</td>
<td>University of Illinois at Urbana Champaign</td>
<td>1999</td>
</tr>
<tr>
<td>BE</td>
<td>CS</td>
<td>TsingHua University</td>
<td>1995</td>
</tr>
<tr>
<td>BE</td>
<td>Business</td>
<td>TsingHua University</td>
<td>1995</td>
</tr>
</tbody>
</table>

4. If you do not have an advanced degree in the program area, describe any course work you may have taken, or other ways in which you have achieved competence in the program area; there is no necessity to repeat information here that is contained in later sections of this document.

NA

5. Conferences, workshops, and professional development programs in which you have participated in the last 5 years to improve teaching and professional competence in the program area:

ACM Mobicom, ACM Mobihoc, IEEE INFOCOM, IEEE ICC, ACM DialM Workshop, COCOON, AAIM

6. Other related computing experience including teaching, industrial, governmental, etc. (where, when, description and scope of duties):

<table>
<thead>
<tr>
<th>Dates</th>
<th>Where</th>
<th>Duties</th>
</tr>
</thead>
</table>

7. Consulting—list agencies and dates, and briefly describe each project:

<table>
<thead>
<tr>
<th>Dates</th>
<th>Agency</th>
<th>Project</th>
</tr>
</thead>
</table>

8. For the academic year in which the Self Study was written list your assigned duties other than for teaching, (committee membership, advising, etc.) with average hours per week. Indicate which, if any, carry extra compensation. If you are course coordinator for courses taught by other than full-time or part time faculty, please indicate here which courses.

Course Coordinator,

9. For the four years preceding the Self Study, list all department, college, and/or university committees of which you are/were a member including year(s) served:

- Graduate Study Committee, 2002-2007
- Graduate admission Committee, 2002-2007

10. Principal publications during the last five years. Give in standard bibliographic format.

1. **Ming-Yang Kao, Xiang-Yang Li, WeiZhao Wang**  
   Average Case Analysis for Binary Tree Labeling Schemes.  

2. **Yu Wang and Weizhao Wang and Xiang-Yang Li**  
   Efficient Distributed Low Cost Backbone Formation for Wireless Networks  

3. **WeiZhao Wang, Xiang-Yang Li, Zheng Sun**  
   Design Differentiated Service Multicast With Selfish Agents  

4. **Weizhao Wang and Xiang-Yang Li**  
   Low-Cost Routing in Selfish and Rational Wireless Ad Hoc Networks,  

5. **Xiang-Yang Li and Yu Wang**  
   Simple Approximation Algorithms and PTASs for Various Problems in Wireless Ad Hoc Networks.  

6. **Wen-Zhan Song and Xiang-Yang Li and Ophir Frieder and WeiZhao Wang**  
   Local Construction of Energy-Efficient and Low-Weighted Topology for Wireless Ad Hoc Networks
7. **Chih-Wei Yi, Peng-Jun Wan, Xiang-Yang Li, Ophir Frieder**, 
Asymptotic Distribution of The Number of Isolated Nodes in Wireless Ad Hoc Networks with Bernoulli Nodes, 

8. **Xiang-Yang Li, Wen-Zhan Song, Yu Wang** 
Localized Topology Control for Heterogenous Wireless Ad Hoc Networks, 

Localized Delaunay Triangulation with Applications in Wireless Ad Hoc Networks 

10. **Xiang-Yang Li, Peng-Jun Wan, Ophir Frieder** 
Coverage Problems in Wireless Ad-hoc Sensor Networks. 

11. **Peng-Jun Wan, G. Calinescu, Xiang-Yang Li, and Ophir Frieder** 
Minimum Energy Broadcast Routing in Static Ad Hoc Wireless Networks 

12. **Xiang-Yang Li, Shao-Jie Tang and Ophir Frieder** 
Multicast Capacity of Large Scale Wireless Ad Hoc Networks 
ACM MobiCom 2007 (Regular). Acceptance ratio: (26 regular +14 extended) / 233

13. **Kiyoko F. Aoki-Kinoshita, Minoru Kanehisa, Ming-Yang Kao, Xiang-Yang Li, and Weizhao Wang** 
A 6-Approximation Algorithm for Computing the Least Common AoN-supertree With Application to the Reconstruction of Glycan Trees

14. **WeiZhao Wang and Yu Wang and Xiang-Yang Li and WenZhan Song and Ophir Frieder** 
Efficient Interference-Aware TDMA Link Scheduling for Static Wireless Networks 
Twelfth Annual International Conference on Mobile Computing and Networking (ACM MobiCom 2006) (Acceptance Ratio: 35 out of 298, 12%)

15. **WeiZhao Wang and Stephan Eidenbez and Yu Wang and Xiang-Yang Li** 
OURS- Optimal Unicast Routing Systems in Non-Cooperative Wireless Networks 
Twelfth Annual International Conference on Mobile Computing and Networking (ACM MobiCom 2006) (Acceptance Ratio: 35 out of 298, 12%)

16. **Xiang-Yang Li and Wen-Zhan Song and WeiZhao Wang** 
A Unified Energy Efficient Topology for Unicast and Broadcast 

17. **Yu Wang and WeiZhao Wang and Xiang-Yang Li** 
Distributed Low-Cost Weighted Backbone Formation for Wireless Ad Hoc Networks 
ACM MobiHoc, 2005. (Acceptance Ratio: 40 out of 281, 14%)
18. Ming-Yang Kao, Xiang-Yang Li, and WeiZhao Wang
(Acceptance Ratio: 33 out of 113, 29%)

19. Xiang-Yang Li and Zheng Sun and WeiZhao Wang
Cost Sharing and Strategyproof Mechanisms for Set Cover Games
(Acceptance Ratio: 54 out of 217, 25%)

20. Weizhao Wang, Xiang-Yang Li, Zheng Sun and Yu Wang
Design Multicast Protocols for Non-Cooperative Networks
(Acceptance Ratio: 244 out of 1419, 15.9%)

21. Weizhao Wang, Xiang-Yang Li and Yu Wang
Truthful Multicast in Selfish Wireless Networks,
(Acceptance Ratio: 26 out of 327, 7.95%)

22. Wen-Zhan Song, Yu Wang, Xiang-Yang Li
Localized Algorithms for Energy Efficient Topology in Wireless Ad Hoc Networks,
(Acceptance Ratio: 24 out of 233, 10.3%)

23. Xiang-Yang Li, Yu Wang, Peng-Jun Wan, Wan-Zhen Song, and Ophir Frieder
Localized Low-Weight Graph and Its Applications in Wireless Ad Hoc Networks,
Accepted for publication, IEEE INFOCOM’2004.
(Acceptance Ratio: 261 out of 1420, 18.3%)

24. Xiang-Yang Li, Yu Wang, Peng-Jun Wan and Chi-Wei Yi
Fault Tolerant Deployment and Topology Control for Wireless Ad Hoc Networks,
(Acceptance Ratio: 29 out of 189, 15.3%)

25. Xiang-Yang Li, G. Calinescu, Peng-Jun Wan
Distributed Construction of Planar Spanner and Routing for Ad Hoc Networks
(Acceptance Ratio: 192 out of 938, 20.5%)

11. Other scholarly activity during the last 5 years: grants, sabbaticals, software development, etc.:

Fall semester 2007, Sabbatical leave, Microsoft Research Asia, BeiJing.

12. Courses taught this and last academic year term-by-term. This year is the academic year in which this Self-Study report is prepared; the last year was the year prior to this. If you were on sabbatical leave, please enter the information for the year prior to the sabbatical. Please list each section of the same course separately.

<table>
<thead>
<tr>
<th>Term/year</th>
<th>Course number</th>
<th>Course title</th>
<th>Semester hrs</th>
<th>Class size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fall 2006</td>
<td>CS430</td>
<td>CS430: Introduction to Algorithms.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Fall 2005</td>
<td>Cs549</td>
<td>Cryptography and Network Security (CS549).</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

13. Estimate the percentage of your time devoted to scholarly and/or research activities: 50%. Please give a brief description of your major research and scholarly activities:

Wireless networking algorithms, keywords auctions, algorithm design and analysis, computational geometry.

14. If you are a part-time faculty member or a full-time faculty member without full-time commitment to the program, state what percentage of full-time you are assigned to the program: ____%.
1. Name, current academic rank, and tenure status:

<table>
<thead>
<tr>
<th>Name:</th>
<th>PENGJUN WAN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rank:</td>
<td>Associate Professor</td>
</tr>
<tr>
<td>Tenure Status:</td>
<td>Tenured</td>
</tr>
</tbody>
</table>

2. Date of original appointment to this faculty, followed by dates and ranks of advancement:

<table>
<thead>
<tr>
<th>Title</th>
<th>Dates Held</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assistant Professor</td>
<td>Aug. 1987-Jul. 2003</td>
</tr>
<tr>
<td>Associate Professor</td>
<td>Aug. 2003-present</td>
</tr>
</tbody>
</table>

3. Degrees with fields, institutions, and dates

<table>
<thead>
<tr>
<th>Degree</th>
<th>Field</th>
<th>Institution</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>PhD</td>
<td>Computer Science</td>
<td>University of Minnesota</td>
<td>May 1997</td>
</tr>
<tr>
<td>MS</td>
<td>Applied Math.</td>
<td>Chinese Academy of Sciences</td>
<td>May 1993</td>
</tr>
<tr>
<td>BS</td>
<td>Applied Math</td>
<td>Tsinghua University</td>
<td>Aug. 2000</td>
</tr>
</tbody>
</table>

4. If you do not have an advanced degree in the program area, describe any course work you may have taken, or other ways in which you have achieved competence in the program area; there is no necessity to repeat information here that is contained in later sections of this document.

NA

5. Conferences, workshops, and professional development programs in which you have participated in the last 5 years to improve teaching and professional competence in the program area:

I served as the Technical Program Committee Co-Chairs of the following conferences and workshops on wireless networking within the last 5 years.
- International Workshop on Theoretical Aspects of Wireless Ad Hoc, Sensor, and Peer-to-Peer Networks (TAWN) 2004. Fully sponsored by NSF.

6. Other related computing experience including teaching, industrial, governmental, etc. 
   (where, when, description and scope of duties):

<table>
<thead>
<tr>
<th>Dates</th>
<th>Where</th>
<th>Duties</th>
</tr>
</thead>
</table>

7. Consulting—list agencies and dates, and briefly describe each project:

<table>
<thead>
<tr>
<th>Dates</th>
<th>Agency</th>
<th>Project</th>
</tr>
</thead>
</table>

8. For the academic year in which the Self Study was written list your assigned duties 
   other than for teaching, (committee membership, advising, etc.) with average hours 
   per week. Indicate which, if any, carry extra compensation. If you are course 
   coordinator for courses taught by other than full-time or part time faculty, please 
   indicate here which courses.

Graduate Admission Committee: 4 hours/week
University Research Council: 1 hour/week
Course coordinator for CS455

9. For the four years preceding the Self Study, list all department, college, and/or 
   university committees of which you are/were a member including year(s) served:

| Graduate Admission Committee: all the past years |
| University Research Council: 2006-present |
| PhD coordinator: 2004 |

10. Principal publications during the last five years. Give in standard bibliographic 
    format.

2. H. Liu, X. Jia, P.-J. Wan, X. Liu, and F. Yao: A Distributed and Efficient Flooding 
    Scheme Using 1-hop Information in Mobile Ad Hoc Networks, IEEE Transactions on 
    April 2007.
    2007.
<table>
<thead>
<tr>
<th>Reference</th>
<th>Title</th>
<th>Journal/Volume/Pages</th>
<th>Year</th>
</tr>
</thead>
</table>

11. Other scholarly activity during the last 5 years: grants, sabbaticals, software development, etc.:
Sabbatical at City University of Hong Kong: Jan. 2005-Dec. 2005

12. Courses taught this and last academic year term-by-term. This year is the academic year in which this Self-Study report is prepared; the last year was the year prior to this. If you were on sabbatical leave, please enter the information for the year prior to the sabbatical. Please list each section of the same course separately.

<table>
<thead>
<tr>
<th>Term/year</th>
<th>Course number</th>
<th>Course title</th>
<th>Semester hrs</th>
<th>Class size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spring/08</td>
<td>CS547</td>
<td>Wireless Networking</td>
<td>3</td>
<td>6</td>
</tr>
<tr>
<td>Fall/07</td>
<td>CS547</td>
<td>Wireless Networking</td>
<td>3</td>
<td>20</td>
</tr>
<tr>
<td>Spring/07</td>
<td>CS547</td>
<td>Wireless Networking</td>
<td>3</td>
<td>14</td>
</tr>
<tr>
<td>Fall/06</td>
<td>CS547</td>
<td>Wireless Networking</td>
<td>3</td>
<td>17</td>
</tr>
</tbody>
</table>

13. Estimate the percentage of your time devoted to scholarly and/or research activities: 70%. Please give a brief description of your major research and scholarly activities:
- Producing research publications,
- Organizing technical conferences and workshops
- Reviewers for journal and conference articles

14. If you are a part time faculty member or a full-time faculty member without full-time commitment to the program, state what percentage of full-time you are assigned to the program: ____%. 

VI - 72
Assistant Professor
1. Name, current academic rank, and tenure status:

<table>
<thead>
<tr>
<th>Name:</th>
<th>Zhiling Lan</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rank:</td>
<td>Assistant Professor</td>
</tr>
<tr>
<td>Tenure Status:</td>
<td>Tenure-track</td>
</tr>
</tbody>
</table>

2. Date of original appointment to this faculty, followed by dates and ranks of advancement:

<table>
<thead>
<tr>
<th>Title</th>
<th>Dates Held</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assistant Professor, IIT CS Dept.</td>
<td>2002 - present</td>
</tr>
</tbody>
</table>

3. Degrees with fields, institutions, and dates

<table>
<thead>
<tr>
<th>Degree</th>
<th>Field</th>
<th>Institution</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ph.D.</td>
<td>Computer Engineering</td>
<td>Northwestern University</td>
<td>2002</td>
</tr>
<tr>
<td>MS</td>
<td>Applied Math</td>
<td>Chinese Academy of Sciences</td>
<td>1995</td>
</tr>
<tr>
<td>BS</td>
<td>Math</td>
<td>Beijing Normal University</td>
<td>1992</td>
</tr>
</tbody>
</table>

4. If you do not have an advanced degree in the program area, describe any course work you may have taken, or other ways in which you have achieved competence in the program area; there is no necessity to repeat information here that is contained in later sections of this document.

5. Conferences, workshops, and professional development programs in which you have participated in the last 5 years to improve teaching and professional competence in the program area:

<table>
<thead>
<tr>
<th>Conference</th>
<th>Years</th>
</tr>
</thead>
<tbody>
<tr>
<td>IEEE Supercomputing</td>
<td>multiple years</td>
</tr>
<tr>
<td>ICPP</td>
<td>multiple years</td>
</tr>
<tr>
<td>IEEE Cluster</td>
<td>multiple years</td>
</tr>
<tr>
<td>IEEE DSN</td>
<td>multiple years</td>
</tr>
</tbody>
</table>

6. Other related computing experience including teaching, industrial, governmental, etc. (where, when, description and scope of duties):

<table>
<thead>
<tr>
<th>Dates</th>
<th>Where</th>
<th>Duties</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
7. Consulting—list agencies and dates, and briefly describe each project:

<table>
<thead>
<tr>
<th>Dates</th>
<th>Agency</th>
<th>Project</th>
</tr>
</thead>
</table>

8. For the academic year in which the Self Study was written list your assigned duties other than for teaching, (committee membership, advising, etc.) with average hours per week. Indicate which, if any, carry extra compensation. If you are course coordinator for courses taught by other than full-time or part time faculty, please indicate here which courses.

| CS Department Undergraduate Studies Committee, 2002 - present |
| University Undergraduate Committee, 2007 - Present           |
| CS Department Graduate Admission Committee, 2006 - present    |

9. For the four years preceding the Self Study, list all department, college, and/or university committees of which you are/were a member including year(s) served:

| Member, CS Department Undergraduate Studies Committee, 2002 - present |
| Member, University Undergraduate Committee, 2007 - Present           |
| Member, CS Department Graduate Admission Committee, 2006 - present    |

10. Principal publications during the last five years. Give in standard bibliographic format.


11. Other scholarly activity during the last 5 years: grants, sabbaticals, software development, etc.:


12. Courses taught this and last academic year term-by-term. This year is the academic year in which this Self-Study report is prepared; the last year was the year prior to this. If you were on sabbatical leave, please enter the information for the year prior to the sabbatical. Please list each section of the same course separately.

<table>
<thead>
<tr>
<th>Term/year</th>
<th>Course number</th>
<th>Course title</th>
<th>Semester hrs</th>
<th>Class size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spring, 2008</td>
<td>CS546-961</td>
<td>Parallel and Distributed Processing</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>Spring, 2008</td>
<td>CS595-02</td>
<td>Fault Tolerance Computing</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>Fall, 2007</td>
<td>CS546</td>
<td>Parallel and Distributed Processing</td>
<td>30</td>
<td></td>
</tr>
<tr>
<td>Spring, 2007</td>
<td>CS550</td>
<td>Advanced Operating Systems</td>
<td>36</td>
<td></td>
</tr>
<tr>
<td>Spring, 2007</td>
<td>CS546-961</td>
<td>Parallel and Distributed Processing</td>
<td>7</td>
<td></td>
</tr>
<tr>
<td>Fall, 2006</td>
<td>CS546</td>
<td>Parallel and Distributed Processing</td>
<td>26</td>
<td></td>
</tr>
</tbody>
</table>

13. Estimate the percentage of your time devoted to scholarly and/or research activities: 80%. Please give a brief description of your major research and scholarly activities:

My research focuses on designing and developing parallel algorithms and software infrastructures to promote efficiency and scalability of large-scale systems and high-performance applications.

http://www.cs.iit.edu/~zlan/research.htm

14. If you are a part time faculty member or a full-time faculty member without full-time commitment to the program, state what percentage of full-time you are assigned to the program: %. 
1. Name, current academic rank, and tenure status:

<table>
<thead>
<tr>
<th>Name:</th>
<th>Shangping Ren</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rank:</td>
<td>Assistant Professor</td>
</tr>
<tr>
<td>Tenure Status:</td>
<td>Not tenured yet (going for tenure review in 2008)</td>
</tr>
</tbody>
</table>

2. Date of original appointment to this faculty, followed by dates and ranks of advancement:

<table>
<thead>
<tr>
<th>Title</th>
<th>Dates Held</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assistant Professor</td>
<td>Aug., 2003</td>
</tr>
</tbody>
</table>

3. Degrees with fields, institutions, and dates

<table>
<thead>
<tr>
<th>Degree</th>
<th>Field</th>
<th>Institution</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ph.D</td>
<td>CS</td>
<td>University of Illinois at Urbana-Champaign</td>
<td>1997</td>
</tr>
</tbody>
</table>

4. If you do not have an advanced degree in the program area, describe any course work you may have taken, or other ways in which you have achieved competence in the program area; there is no necessity to repeat information here that is contained in later sections of this document.

5. Conferences, workshops, and professional development programs in which you have participated in the last 5 years to improve teaching and professional competence in the program area:

6. Other related computing experience including teaching, industrial, governmental, etc. (where, when, description and scope of duties):
7. Consulting—list agencies and dates, and briefly describe each project:

<table>
<thead>
<tr>
<th>Dates</th>
<th>Agency</th>
<th>Project</th>
</tr>
</thead>
<tbody>
<tr>
<td>March, 2007 to</td>
<td>ITT</td>
<td>Research on retrofitting legacy critical infrastructures with cyber</td>
</tr>
<tr>
<td>March 2008</td>
<td></td>
<td>security features</td>
</tr>
</tbody>
</table>

8. For the academic year in which the Self Study was written list your assigned duties other than for teaching, (committee membership, advising, etc.) with average hours per week. Indicate which, if any, carry extra compensation. If you are course coordinator for courses taught by other than full-time or part time faculty, please indicate here which courses.

9. For the four years preceding the Self Study, list all department, college, and/or university committees of which you are/were a member including year(s) served:

10. Principal publications during the last five years. Give in standard bibliographic format.

Satisfaction Probabilities of Interval-based Timing Constraints Yue Yu, Shangping Ren and Ophir Frieder, IEEE Transactions on Computers (Accepted)

Science (To be published in Feb., 2008)


The Role of Roles for Supporting Reconfigurability in Open Distributed and Embedded Systems Shangping Ren, Nianen Chen, Yue Yu, Jeffrey Tsai and Kevin Kwiat, ACM Transactions on Autonomous and Adaptive Systems, Volume 2, Number 3, 2007


Actors, Roles and Coordinators — A Coordination Model for Open Distributed Embedded Systems Shangping Ren, Yue Yu, Nianen Chen, Kevin Marth, Pierre-Etienne Poirot, and Limin Shen, Proceedings of the 8th IFIP International Conference on Coordination Models and Languages, LNCS Vol 4038, 2006


11. Other scholarly activity during the last 5 years: grants, sabbaticals, software development, etc.:

CAREER: Behavior-Based Coordination for Open Distributed Real-Time and Embedded Applications
Funding Agent: National Science Foundation
Amount and Period: $400,000, 9/2008 – 8/2013

Support for Organizing First International Workshop on Physical Cyber Systems
Funding Agent: National Science Foundation

Research Experience for Undergraduates Supplement: Modeling and Programming Distributed Adaptive Real-Time Embedded Systems
Funding Agent: National Science Foundation

Modeling and Programming Distributed Adaptive Real-Time Embedded Systems
Funding Agent: National Science Foundation

A Self-Diagnosing, Configurable Distributed System for Testing Superconducting Accelerator Magnets
Funding Agent: Fermi National Accelerator Laboratory
12. Courses taught this and last academic year term-by-term. This year is the academic year in which this Self-Study report is prepared; the last year was the year prior to this. If you were on sabbatical leave, please enter the information for the year prior to the sabbatical. Please list each section of the same course separately.

<table>
<thead>
<tr>
<th>Term/year</th>
<th>Course number</th>
<th>Course title</th>
<th>Semester hrs</th>
<th>Class size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fall 2003</td>
<td>Cs595</td>
<td>Special topics in distributed and real-time systems</td>
<td>3</td>
<td>~ 10</td>
</tr>
<tr>
<td>Spring 2004</td>
<td>Cs521</td>
<td>Object Oriented Design</td>
<td>3</td>
<td>~ 20</td>
</tr>
<tr>
<td>Fall 2004</td>
<td>Cs447</td>
<td>Distributed Objects</td>
<td>3</td>
<td>~ 20</td>
</tr>
<tr>
<td>Spring 2005</td>
<td>Cs447</td>
<td>Distributed Objects</td>
<td>3</td>
<td>~ 10</td>
</tr>
<tr>
<td>Fall 2005</td>
<td>Cs330</td>
<td>Discrete Structures</td>
<td>3</td>
<td>~ 20</td>
</tr>
<tr>
<td>Spring 2006</td>
<td>Cs 447</td>
<td>Distributed Objects</td>
<td>3</td>
<td>~ 10</td>
</tr>
<tr>
<td>Fall 2006</td>
<td>Cs540</td>
<td>Programming Language Design</td>
<td>3</td>
<td>~ 10</td>
</tr>
<tr>
<td>Spring 2007</td>
<td>Cs595</td>
<td>Special topics in distributed and real-time embedded systems</td>
<td>3</td>
<td>~ 10</td>
</tr>
<tr>
<td>Fall 2007</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Spring 2008</td>
<td>Cs695</td>
<td>Doctoral Seminar</td>
<td>1</td>
<td>~ 10</td>
</tr>
</tbody>
</table>

13. Estimate the percentage of your time devoted to scholarly and/or research activities: ___100___%. Please give a brief description of your major research and scholarly activities:

Teaching and research in the field of distributed, real-time and embedded systems, cyber physical systems; advising phd students; journal/conference paper reviews; organizing conference/workshops in the related area.

14. If you are a part time faculty member or a full-time faculty member without full-time commitment to the program, state what percentage of full-time you are assigned to the program: ______%.
1. Name, current academic rank, and tenure status:

<table>
<thead>
<tr>
<th>Name:</th>
<th>Wai Gen Yee</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rank:</td>
<td>Assistant Professor</td>
</tr>
<tr>
<td>Tenure Status:</td>
<td>Untenured</td>
</tr>
</tbody>
</table>

2. Date of original appointment to this faculty, followed by dates and ranks of advancement:

<table>
<thead>
<tr>
<th>Title</th>
<th>Dates Held</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assistant Professor</td>
<td>8/2003 – present</td>
</tr>
</tbody>
</table>

3. Degrees with fields, institutions, and dates

<table>
<thead>
<tr>
<th>Degree</th>
<th>Field</th>
<th>Institution</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>PhD</td>
<td>CS</td>
<td>Georgia Institute of Technology</td>
<td>8/2003</td>
</tr>
<tr>
<td>MS</td>
<td>CS</td>
<td>Georgia Institute of Technology</td>
<td>8/2000</td>
</tr>
<tr>
<td>MS</td>
<td>Math</td>
<td>University of Chicago</td>
<td>6/1995</td>
</tr>
</tbody>
</table>

4. If you do not have an advanced degree in the program area, describe any course work you may have taken, or other ways in which you have achieved competence in the program area; there is no necessity to repeat information here that is contained in later sections of this document.

5. Conferences, workshops, and professional development programs in which you have participated in the last 5 years to improve teaching and professional competence in the program area:

- ACM SIGMOD Conference, 2006
- IEEE ICDE Conference, 2006
- IEEE IPDPS Conference, 2007

6. Other related computing experience including teaching, industrial, governmental, etc. (where, when, description and scope of duties):

<table>
<thead>
<tr>
<th>Dates</th>
<th>Where</th>
<th>Duties</th>
</tr>
</thead>
</table>
7. Consulting—list agencies and dates, and briefly describe each project:

<table>
<thead>
<tr>
<th>Dates</th>
<th>Agency</th>
<th>Project</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

8. For the academic year in which the Self Study was written list your assigned duties other than for teaching, (committee membership, advising, etc.) with average hours per week. Indicate which, if any, carry extra compensation. If you are course coordinator for courses taught by other than full-time or part-time faculty, please indicate here which courses.

| CS Department Undergraduate Studies Committee (2hrs/wk) |                                                                        |
| University Faculty Council (2hrs/wk)                   |                                                                        |
| College of Science and Letters Dean Search Committee (50 hours total) |                                                                        |
| Student advisement (2hrs/week)                         |                                                                        |
| CS Department Employment Seminar (5 hours total per semester) |                                                                        |

9. For the four years preceding the Self Study, list all department, college, and/or university committees of which you are/were a member including year(s) served:

| University Undergraduate Studies Committee (2006-7)       |                                                                        |
| Research Dean Evaluation Committee (2006)                |                                                                        |
| CS Department Undergraduate Studies Committee (2hrs/wk)  |                                                                        |
| Student advisement (2hrs/week)                           |                                                                        |
| CS Department Employment Seminar (5 hours total per semester) |                                                                        |

10. Principal publications during the last five years. Give in standard bibliographic format.

| Wai Gen Yee, Linh Thai Nguyen, Ophir Frieder: Masked Queries for Search Accuracy in Peer-to-Peer File-Sharing Systems. IPDPS 2007: 1-10 |                                                                        |
| Wai Gen Yee, Linh Thai Nguyen, Ophir Frieder: Masked Queries for Search Accuracy in Peer-to-Peer File-Sharing Systems. IPDPS 2007: 1-10 |                                                                        |
| Linh Thai Nguyen, Dongmei Jia, Wai Gen Yee, Ophir Frieder: An analysis of peer-to-peer file-sharing system queries. SIGIR 2007: 855-856 |                                                                        |
| Linh Thai Nguyen, Dongmei Jia, Wai Gen Yee, Ophir Frieder: An analysis of peer-to-peer file-sharing system queries. SIGIR 2007: 855-856 |                                                                        |
| Ying Lai, Ratko Orlandic, Wai Gen Yee, Sachin Kulkarni: Scalable Clustering for Large High-Dimensional Data Based on Data Summarization. CIDM 2007: 456-461 |                                                                        |
11. Other scholarly activity during the last 5 years: grants, sabbaticals, software development, etc.:

- Establishment of the Information Security Center, which has received NSA Center for Academic Excellence Certification (2006, renewed 2008, with D. Grossman)
- Co-organizer, ACM Workshop on Open Source Information Retrieval, 2005-2006 (with M. Beigbeder, EMSE and W. Buntine, HIIT)
- Co-organizer, the IEEE ICDE Ph.D. Workshop, 2006 (with S. Navathe, Georgia Tech)
- Co-organizer, the Midwest Workshop on Curriculum Development in Information Assurance and Security, 2005 (with D. Grossman, O. Frieder, J. Montgomery, DePaul Univ.)

12. Courses taught this and last academic year term-by-term. This year is the academic year in which this Self-Study report is prepared; the last year was the year prior to this. If you were on sabbatical leave, please enter the information for the year prior to the sabbatical. Please list each section of the same course separately.

<table>
<thead>
<tr>
<th>Term/year</th>
<th>Course number</th>
<th>Course title</th>
<th>Semester hrs</th>
<th>Class size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spr/2008</td>
<td>CS525</td>
<td>Advanced Database Organization</td>
<td>3</td>
<td>40</td>
</tr>
<tr>
<td>Spr/2008</td>
<td>IPRO313</td>
<td>Ultra-High Speed Ticker System</td>
<td>3</td>
<td>10</td>
</tr>
<tr>
<td>Aut/2007</td>
<td>CS525</td>
<td>Advanced Database Organization</td>
<td>3</td>
<td>40</td>
</tr>
<tr>
<td>Aut/2007</td>
<td>IPRO313</td>
<td>Ultra-High Speed Ticker System</td>
<td>3</td>
<td>10</td>
</tr>
<tr>
<td>Aut/2007</td>
<td>CS495</td>
<td>Topics in Information Retrieval</td>
<td>3</td>
<td>6</td>
</tr>
<tr>
<td>Sum/2007</td>
<td>CS525</td>
<td>Advanced Database Organization</td>
<td>3</td>
<td>6</td>
</tr>
<tr>
<td>Spr/2007</td>
<td>CS525</td>
<td>Advanced Database Organization</td>
<td>3</td>
<td>40</td>
</tr>
<tr>
<td>Spr/2007</td>
<td>CS495</td>
<td>Educational Projects in Computer Science</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>Aut/2006</td>
<td>CS525</td>
<td>Advanced Database Organization</td>
<td>3</td>
<td>40</td>
</tr>
<tr>
<td>Aut/2006</td>
<td>CS595</td>
<td>Topics in Distributed Information Systems</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>Sum/2006</td>
<td>CS525</td>
<td>Advanced Database Organization</td>
<td>3</td>
<td>6</td>
</tr>
</tbody>
</table>

13. Estimate the percentage of your time devoted to scholarly and/or research activities: **75%**. Please give a brief description of your major research and scholarly activities:

I study search engines and distributed systems. I spent the last several years developing peer-to-peer search engines. I am also starting research on high-performance search engines and query processing. I am on several program committees in search and distributed systems. I have also organized several workshops on these areas.

14. If you are a part time faculty member or a full-time faculty member without full-time commitment to the program, state what percentage of full-time you are assigned to the program: ______%.
Industry/Clinical Professor
1. Name, current academic rank, and tenure status:

<table>
<thead>
<tr>
<th>Name:</th>
<th>Edward Chlebus</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rank:</td>
<td>Industry Associate Professor</td>
</tr>
<tr>
<td>Tenure Status:</td>
<td>Non-TT position</td>
</tr>
</tbody>
</table>

2. Date of original appointment to this faculty, followed by dates and ranks of advancement:

<table>
<thead>
<tr>
<th>Title</th>
<th>Dates Held</th>
</tr>
</thead>
<tbody>
<tr>
<td>Senior Lecturer</td>
<td>2001-2005</td>
</tr>
<tr>
<td>Industry Associate Professor</td>
<td>2005 – present</td>
</tr>
</tbody>
</table>

3. Degrees with fields, institutions, and dates

<table>
<thead>
<tr>
<th>Degree</th>
<th>Field</th>
<th>Institution</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>MS</td>
<td>Electrical Engineering</td>
<td>Cracow University, Poland</td>
<td>1985</td>
</tr>
<tr>
<td>PhD</td>
<td>Electrical Engineering</td>
<td>Cracow University, Poland</td>
<td>1990</td>
</tr>
</tbody>
</table>

4. If you do not have an advanced degree in the program area, describe any course work you may have taken, or other ways in which you have achieved competence in the program area; there is no necessity to repeat information here that is contained in later sections of this document.

5. Conferences, workshops, and professional development programs in which you have participated in the last 5 years to improve teaching and professional competence in the program area:

- 18th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, PIMRC’07, Athens, Greece, Sept. 2007
- IEEE Wireless Communications and Networking Conference, WCNC’07, Hong Kong, March 2007
- IEEE Global Communications Conference, GLOBECOM’05, St. Louis, MO, Nov. 2005
- Motivating Unmotivated Students: A Teaching Enrichment Workshop, workshop organized by IIT College of Science and Letters, Chicago, IL, Nov. 2005
- Interactive Teaching Methods: A Teaching Enrichment Workshop, workshop organized by IIT College of Science and Letters, Chicago, IL, April 2004
- Active Student Involvement Through Active Teaching, workshop organized by IIT
6. Other related computing experience including teaching, industrial, governmental, etc. (where, when, description and scope of duties):

<table>
<thead>
<tr>
<th>Dates</th>
<th>Where</th>
<th>Duties</th>
</tr>
</thead>
<tbody>
<tr>
<td>1988-1989</td>
<td>Laboratoire d’Automatique et d’Analyse des Systemes du CNRS, Toulouse, France</td>
<td>Visiting Scholar; design of adaptive call routing schemes for the Paris metro area network; research sponsored by France Telecom R&amp;D Center in Paris</td>
</tr>
<tr>
<td>1990-1992</td>
<td>Telecommunications Department, Cracow University, Poland</td>
<td>Assistant Professor; teaching and research, consulting with Polish PTT</td>
</tr>
<tr>
<td>1992-Dec. 1994</td>
<td>Teletraffic Research Centre, University of Adelaide, Australia</td>
<td>Research Fellow, teletraffic engineering of cellular mobile systems; research sponsored by Telecom Australia Research Labs in Melbourne</td>
</tr>
<tr>
<td>Jan. 1995-1997</td>
<td>Motorola Satellite Communications Division, Chandler, AZ</td>
<td>Project leader of the Network Integrity Support Team; performance analysis of mobile satellite communications systems</td>
</tr>
<tr>
<td>1997-2001</td>
<td>Bell Labs, Lucent Technologies, Naperville, IL</td>
<td>Member of Technical Staff; research and development of cellular and PCS systems</td>
</tr>
</tbody>
</table>

7. Consulting—list agencies and dates, and briefly describe each project:

<table>
<thead>
<tr>
<th>Dates</th>
<th>Agency</th>
<th>Project</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sept. 2006</td>
<td>Nokia Research Center, San Diego, CA</td>
<td>Review of IEEE 802.11n Standard</td>
</tr>
<tr>
<td>2006-2008</td>
<td>Azure Wireless, Melbourne, Australia</td>
<td>Analysis of session traffic for Wi-Fi wireless Internet access</td>
</tr>
</tbody>
</table>

8. For the academic year in which the Self Study was written list your assigned duties other than for teaching, (committee membership, advising, etc.) with average hours per week. Indicate which, if any, carry extra compensation. If you are course coordinator for courses taught by other than full-time or part time faculty, please indicate here which courses.
CS Dept. Graduate Admission Committee – 7 hours/week
Advising – 5 hours/week
Scholarship coordinator – 1 hour/week

9. For the four years preceding the Self Study, list all department, college, and/or university committees of which you are/were a member including year(s) served:

10. Principal publications during the last five years. Give in standard bibliographic format.

| Divgi G., Chlebus E., The impact of Internet access account categories on user and traffic in a commercial nationwide Wi-Fi hotspot network, Proc. 5th IEEE Consumer Communications and Networking Conference, CCNC’08, Las Vegas, NV, Jan. 2008 |
| Divgi G., Chlebus E., User and traffic characteristics of a commercial nationwide Wi-Fi hotspot network, Proc. 18th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, PIMRC’07, Athens, Greece, Sept. 2007 |
| Chlebus E., Ohri R., Estimating parameters of the Pareto distribution by means of Zipf’s law: Application to Internet research, Proc. IEEE Global Communications Conference, GLOBECOM’05, St. Louis, MO, Nov. 2005 |

11. Other scholarly activity during the last 5 years: grants, sabbaticals, software development, etc.:

| Conference session chair (IEEE Globecom, IEEE PIMRC, IEEE WCNC) |
| Frequent book, journal and conference reviewer |
| IEEE investigator of a plagiarism case |
| Editor of the Open Information Systems Journal |
| Conference technical program committee member |
| Grant proposer |
12. Courses taught this and last academic year term-by-term. This year is the academic year in which this Self-Study report is prepared; the last year was the year prior to this. If you were on sabbatical leave, please enter the information for the year prior to the sabbatical. Please list each section of the same course separately.

<table>
<thead>
<tr>
<th>Term/year</th>
<th>Course number</th>
<th>Course title</th>
<th>Semester hrs</th>
<th>Class size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fall 06</td>
<td>CS-542</td>
<td>Computer Networks I</td>
<td>3</td>
<td>85</td>
</tr>
<tr>
<td></td>
<td>CS-542-051</td>
<td></td>
<td>3</td>
<td>35</td>
</tr>
<tr>
<td></td>
<td>CS-542-092</td>
<td></td>
<td>3</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>CS-542-251</td>
<td></td>
<td>3</td>
<td>30</td>
</tr>
<tr>
<td></td>
<td>CS-542-961</td>
<td></td>
<td>3</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>CS-544</td>
<td>Computer Networks II</td>
<td>3</td>
<td>31</td>
</tr>
<tr>
<td></td>
<td>CS-544-051</td>
<td></td>
<td>3</td>
<td>21</td>
</tr>
<tr>
<td></td>
<td>CS-544-092</td>
<td></td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>CS-544-251</td>
<td></td>
<td>3</td>
<td>8</td>
</tr>
<tr>
<td>Spring 07</td>
<td>CS-544-051</td>
<td>Computer Networks II</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>CS-555</td>
<td>Analytic Models and Simulation of Computer Systems</td>
<td>3</td>
<td>22</td>
</tr>
<tr>
<td></td>
<td>CS-555-051</td>
<td></td>
<td>3</td>
<td>7</td>
</tr>
<tr>
<td></td>
<td>CS-555-251</td>
<td></td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>CS-555-961</td>
<td></td>
<td>3</td>
<td>12</td>
</tr>
<tr>
<td>Fall 07</td>
<td>CS-542</td>
<td>Computer Networks I</td>
<td>3</td>
<td>114</td>
</tr>
<tr>
<td></td>
<td>CS-542-051</td>
<td></td>
<td>3</td>
<td>52</td>
</tr>
<tr>
<td></td>
<td>CS-542-052</td>
<td></td>
<td>3</td>
<td>31</td>
</tr>
<tr>
<td></td>
<td>CS-542-251</td>
<td></td>
<td>3</td>
<td>9</td>
</tr>
<tr>
<td></td>
<td>CS-542-961</td>
<td></td>
<td>3</td>
<td>22</td>
</tr>
</tbody>
</table>

13. Estimate the percentage of your time devoted to scholarly and/or research activities: ___60__%. Please give a brief description of your major research and scholarly activities:

Network modeling, teletraffic engineering, Internet workload analysis, performance evaluation of computer and communications systems (research, reviewing books and journal/conference papers, advising PhD/MS students, preparing grant proposals (see Section 11))

14. If you are a part time faculty member or a full-time faculty member without full-time commitment to the program, state what percentage of full-time you are assigned to the program: _____%.
1. Name, current academic rank, and tenure status:

<table>
<thead>
<tr>
<th>Name:</th>
<th>Nazli Goharian</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rank:</td>
<td>Clinical Associate Professor</td>
</tr>
<tr>
<td>Tenure Status:</td>
<td>Non Tenure track</td>
</tr>
</tbody>
</table>

2. Date of original appointment to this faculty, followed by dates and ranks of advancement:

<table>
<thead>
<tr>
<th>Title</th>
<th>Dates Held</th>
</tr>
</thead>
<tbody>
<tr>
<td>Senior Lecturer</td>
<td>August 2000 – July 2001</td>
</tr>
<tr>
<td>Clinical Assistant Professor</td>
<td>August 2001 – September 2006</td>
</tr>
<tr>
<td>Clinical Associate Professor</td>
<td>September 2006 – Present</td>
</tr>
</tbody>
</table>

3. Degrees with fields, institutions, and dates

<table>
<thead>
<tr>
<th>Degree</th>
<th>Field</th>
<th>Institution</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>PhD</td>
<td>Computer Science</td>
<td>Florida Institute of Technology</td>
<td>July 2001</td>
</tr>
<tr>
<td>MS</td>
<td>Computer Science</td>
<td>George Mason University</td>
<td>May 1995</td>
</tr>
<tr>
<td>BS</td>
<td>Computer Science</td>
<td>Dortmund University, Germany</td>
<td>May 1992</td>
</tr>
</tbody>
</table>

4. If you do not have an advanced degree in the program area, describe any course work you may have taken, or other ways in which you have achieved competence in the program area; there is no necessity to repeat information here that is contained in later sections of this document.

N/A

5. Conferences, workshops, and professional development programs in which you have participated in the last 5 years to improve teaching and professional competence in the program area:

- ACM Infoscale Conference, 2008
- SIAM Conference on Data Mining (SDM 2008) - Accepted paper & Presentation
- IEEE 5th International Conference on Intelligence and Security Informatics (ISI), 2007. - Accepted paper & Presentation
- ACM 16th Conference in Information and Knowledge Management (CIKM), 2007
- ACM SIGIR (Special Interest Group in Information Retrieval), 2007
<table>
<thead>
<tr>
<th>ACM SIGIR (Special Interest Group in Information Retrieval)</th>
<th>2006</th>
</tr>
</thead>
<tbody>
<tr>
<td>Text Retrieval Conference (Trec), 2006 – Research work participation</td>
<td></td>
</tr>
<tr>
<td>IEEE 4th International Conference on Intelligence and Security Informatics (ISI), 2006. Accepted paper &amp; Presentation</td>
<td></td>
</tr>
<tr>
<td>ACM 14th Conference on Information and Knowledge Management (CIKM), November 2005 - Accepted paper &amp; Presentation</td>
<td></td>
</tr>
<tr>
<td>ACM SIGIR (Special Interest Group in Information Retrieval), 2004</td>
<td></td>
</tr>
<tr>
<td>ACM 13th Conference on Information and Knowledge Management (CIKM), 2004. Accepted paper &amp; Presentation</td>
<td></td>
</tr>
<tr>
<td>ACM 13th Conference on Information and Knowledge Management (CIKM), 2003. Accepted paper &amp; Presentation</td>
<td></td>
</tr>
</tbody>
</table>

6. Other related computing experience including teaching, industrial, governmental, etc. (where, when, description and scope of duties):

<table>
<thead>
<tr>
<th>Dates</th>
<th>Where</th>
<th>Duties</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

7. Consulting—list agencies and dates, and briefly describe each project:

<table>
<thead>
<tr>
<th>Dates</th>
<th>Agency</th>
<th>Project</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

8. For the academic year in which the Self Study was written list your assigned duties other than for teaching, (committee membership, advising, etc.) with average hours per week. Indicate which, if any, carry extra compensation. If you are course coordinator for courses taught by other than full-time or part time faculty, please indicate here which courses.

<table>
<thead>
<tr>
<th>Member, Undergraduate Curriculum Committee &amp; ABET Accreditation (2001- Present)</th>
<th>2 hours/week</th>
</tr>
</thead>
<tbody>
<tr>
<td>Member, Graduate Admissions Committee (2000- Present)</td>
<td>2 hours/week</td>
</tr>
<tr>
<td>Advising graduate students</td>
<td>2 hours/week</td>
</tr>
<tr>
<td>Students research supervision</td>
<td>6 hours/week</td>
</tr>
</tbody>
</table>

9. For the four years preceding the Self Study, list all department, college, and/or university committees of which you are/were a member including year(s) served:
10. Principal publications during the last five years. Give in standard bibliographic format.

**2003**


**2004**


**2005**


Conference on Information and Knowledge Management (CIKM), November 2005.

2006

2007

2008

TREC

11. Other scholarly activity during the last 5 years: grants, sabbaticals, software development, etc.:

Grants
"Enhancing the CS Undergraduate Curriculum to Include Data Mining and Information Retrieval" $339,500
National Science Foundation Award # 0119469, Total of 4 participants, Co-PI.

"CRCD/EI: An Undergraduate CS Specialization in Knowledge Management Systems" $281,999
National Science Foundation, Oct 2004-Sept 2007, Total of 5 participants, Co-PI.

"Biomedical Literature Search" $28,763
IIT Research Institute, September 2005-August 2006, Principle Investigator (PI).

Educational Tools Developments:
Developed the following educational tools for two of the CS classes I developed. These instructional software tools have shown to help the students to learn the material in greater depth.
- IRIS (CS429)
- SimpleDM (CS422)

12. Courses taught this and last academic year term-by-term. This year is the academic year in which this Self-Study report is prepared; the last year was the year prior to this. If you were on sabbatical leave, please enter the information for the year prior to the sabbatical. Please list each section of the same course separately.

<table>
<thead>
<tr>
<th>Term/year</th>
<th>Course number</th>
<th>Course title</th>
<th>Semester hrs</th>
<th>Class size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spring 08</td>
<td>CS425</td>
<td>Database Organizations</td>
<td>3</td>
<td>56</td>
</tr>
<tr>
<td>Spring 08</td>
<td>CS422</td>
<td>Data Mining</td>
<td>3</td>
<td>33</td>
</tr>
<tr>
<td>Spring 08</td>
<td>CS595</td>
<td>Scalable. IR &amp; Knowl. Discovery</td>
<td>3</td>
<td>7</td>
</tr>
<tr>
<td>Fall 07</td>
<td>CS425</td>
<td>Database Organizations</td>
<td>3</td>
<td>47</td>
</tr>
<tr>
<td>Fall 07</td>
<td>CS429</td>
<td>Information Retrieval</td>
<td>3</td>
<td>16</td>
</tr>
<tr>
<td>Fall 07</td>
<td>CS595</td>
<td>Topics in Information Retrieval</td>
<td>3</td>
<td>6</td>
</tr>
<tr>
<td>Spring 07</td>
<td>CS425</td>
<td>Database Organizations</td>
<td>3</td>
<td>53</td>
</tr>
<tr>
<td>Spring 07</td>
<td>CS422</td>
<td>CS422 – Data Mining</td>
<td>3</td>
<td>24</td>
</tr>
<tr>
<td>Spring 07</td>
<td>IPRO</td>
<td></td>
<td>3</td>
<td>13</td>
</tr>
<tr>
<td>Fall 06</td>
<td>CS425</td>
<td>Database Organizations</td>
<td>3</td>
<td>50</td>
</tr>
<tr>
<td>Fall 06</td>
<td>CS422</td>
<td>Information Retrieval</td>
<td>3</td>
<td>18</td>
</tr>
</tbody>
</table>

13. Estimate the percentage of your time devoted to scholarly and/or research activities: 30_______%. Please give a brief description of your major research and scholarly activities:
Conference Committee Participation Role
Program Committee, ACM Conference on Information & Knowledge Management (CIKM), 2001-2006, 2008
Program Committee, IEEE Int’l Conference on Scalable Information Systems (INFOSCALE), 2006-2008
Program Committee, IEEE conference on String Processing in Information Retrieval (SPIRE), 2006
Program Committee, ACM SIGIR Workshop on Open Source Information Retrieval (OSIR), 2006
Program Committee, IEEE Int’l Conference on Intelligence and Security Informatics (ISI), 2007-2008

Refereeing:
IEEE Transactions of Computers
Journal of the American Society for Information Science and Technology
The Computer Journal
Information Processing Letters
Journal of Computer Systems Science & Engineering
Wiley Publisher: Database book
Addison Wesley: Introduction to data Mining; by Pan-Ning Tan, Michael Steinbach, Vipin Kumar

Research Projects (Primary faculty on the projects)
1. Text Classification (on-going)
With the ever-increasing number of digital documents, the ability to automatically classify those documents both quickly and accurately is becoming more critical and difficult. We present Fast Algorithm for Categorizing Text (FACT), which is a statistical based multi-way classifier with our proposed feature selection, Ambiguity Measure (AM), which uses only the most unambiguous keywords to predict the category of a document. Our empirical results show that FACT outperforms the best results on the best performing feature selection for the Naive Bayes classifier namely, Odds Ratio. We empirically show the effectiveness of our approach in outperforming Odds Ratio using four benchmark datasets with a statistical significance of 99% confidence level. Furthermore, the performance of FACT is comparable or better than current non-statistical based classifiers.

2. Detecting Misuse of Information Retrieval Systems (on-going)
Authorized users electronically accessing, potentially via a search engine, documents that are “off-topic” to their pre-defined area of interest is of concern to every private, commercial, and governmental organization. Detecting such misuse, via information retrieval techniques, is the focus of our work.
The topic of misuse detection, particularly as it relates to information retrieval systems, is only of recent interest. Jointly with several students, I presented approaches based on information retrieval processing techniques to detect the misuse of information retrieval systems. These algorithms rely on relevance feedback, document clustering, query results
clustering, or a fusion thereof to detect off topic queries or result sets. We investigated the affect of user query sequences, as well as query length. Meanwhile we proposed a classification-based method using training data.

**Research Projects (Secondary faculty on the project)**

3. **Biomedical Literature Search (on-going)**

Searching biomedical literature differs from conventional search in that the vocabulary (terms) involve significantly different grammatical structures. Suffixes vary in nature; synonyms are far more common; the reliance of taxonomies is far greater. Together with a doctoral student, I participated in the Genomics Track of the international “bake-off” for text competition (TREC) sponsored by the National Institute of Standards and Technology (NIST) and achieved competitive scores – statistically equivalent to those of the top teams. Over 40 teams from leading universities worldwide participated.

14. If you are a part time faculty member or a full-time faculty member without full-time commitment to the program, state what percentage of full-time you are assigned to the program: _____%.
1. Name, current academic rank, and tenure status:

<table>
<thead>
<tr>
<th>Name:</th>
<th>Francis Leung</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rank:</td>
<td>Industry Professor</td>
</tr>
<tr>
<td>Tenure Status:</td>
<td>Non tenure track</td>
</tr>
</tbody>
</table>

2. Date of original appointment to this faculty, followed by dates and ranks of advancement:

<table>
<thead>
<tr>
<th>Title</th>
<th>Dates Held</th>
</tr>
</thead>
<tbody>
<tr>
<td>Industry Professor</td>
<td>Since Fall, 2006</td>
</tr>
</tbody>
</table>

3. Degrees with fields, institutions, and dates

<table>
<thead>
<tr>
<th>Degree</th>
<th>Field</th>
<th>Institution</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>PhD</td>
<td>CS</td>
<td>University of California, Berkeley</td>
<td>1979</td>
</tr>
<tr>
<td>MS</td>
<td>CS</td>
<td>University of California, Berkeley</td>
<td>1975</td>
</tr>
<tr>
<td>BS</td>
<td>EE</td>
<td>California State University at Northridge</td>
<td>1974</td>
</tr>
</tbody>
</table>

4. If you do not have an advanced degree in the program area, describe any course work you may have taken, or other ways in which you have achieved competence in the program area; there is no necessity to repeat information here that is contained in later sections of this document.

5. Conferences, workshops, and professional development programs in which you have participated in the last 5 years to improve teaching and professional competence in the program area:

<table>
<thead>
<tr>
<th>Conference</th>
</tr>
</thead>
<tbody>
<tr>
<td>10th International Conference on High Assurance System Engineering, November, 2007</td>
</tr>
<tr>
<td>9th International Conference on Feature Interaction, September, 2007</td>
</tr>
<tr>
<td>National Workshop on Composable and Systems Technology for High Confidence Cyber-Physical Systems, July, 2007</td>
</tr>
<tr>
<td>NSF Workshop on Cyber-Physical Systems, November, 2006</td>
</tr>
<tr>
<td>InfoScale, June, 2006</td>
</tr>
<tr>
<td>International Conference on Networking and Services, October, 2005</td>
</tr>
<tr>
<td>The 9th World Multiconference on Systematics, Cybernetics and Informatics, July, 2005</td>
</tr>
<tr>
<td>International Conference on Feature Interaction, June, 2005</td>
</tr>
<tr>
<td>IEEE Sarnoff Symposium, April, 2005</td>
</tr>
<tr>
<td>Aspect Oriented Software Development, March, 2005</td>
</tr>
</tbody>
</table>
6. Other related computing experience including teaching, industrial, governmental, etc. (where, when, description and scope of duties):

<table>
<thead>
<tr>
<th>Dates</th>
<th>Where</th>
<th>Duties</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fall, 2003</td>
<td>EECS Dept, IIT</td>
<td>Research Professor</td>
</tr>
<tr>
<td>2001/2003</td>
<td>Nuwave Software</td>
<td>Co-founder, software tools</td>
</tr>
<tr>
<td>1995/2001</td>
<td>Motorola</td>
<td>Senior Resource Manager, GSM cellular network architecture and product roadmap</td>
</tr>
<tr>
<td>1978/1995</td>
<td>Bell Laboratories</td>
<td>Technical manager, software development, forward looking work.</td>
</tr>
</tbody>
</table>

7. Consulting—list agencies and dates, and briefly describe each project:

<table>
<thead>
<tr>
<th>Dates</th>
<th>Agency</th>
<th>Project</th>
</tr>
</thead>
<tbody>
<tr>
<td>2001/2003</td>
<td>TRA</td>
<td>GSM, Multimedia communications</td>
</tr>
</tbody>
</table>

8. For the academic year in which the Self Study was written list your assigned duties other than for teaching, (committee membership, advising, etc.) with average hours per week. Indicate which, if any, carry extra compensation. If you are course coordinator for courses taught by other than full-time or part time faculty, please indicate here which courses:

9. For the four years preceding the Self Study, list all department, college, and/or university committees of which you are/were a member including year(s) served:

10. Principal publications during the last five years. Give in standard bibliographic format.


   With L. Nadeau, “Message Ordered Multicast with Common Building Blocks,” InfoScale
2006, June, 2006


“Writing Reusable Feature Programs with the Feature Language Extensions,” The 8th International Conference on Feature Interaction, June, 2005.


11. Other scholarly activity during the last 5 years: grants, sabbaticals, software development, etc.:

12. Courses taught this and last academic year term-by-term. This year is the academic year in which this Self-Study report is prepared; the last year was the year prior to this. If you were on sabbatical leave, please enter the information for the year prior to the sabbatical. Please list each section of the same course separately.

<table>
<thead>
<tr>
<th>Term/year</th>
<th>Course number</th>
<th>Course title</th>
<th>Semester hrs</th>
<th>Class size</th>
</tr>
</thead>
<tbody>
<tr>
<td>S/2008</td>
<td>551</td>
<td>Design and Implementation of Operating Systems</td>
<td>3</td>
<td>30</td>
</tr>
<tr>
<td>S/2008</td>
<td>487</td>
<td>Software Engineering</td>
<td>3</td>
<td>40</td>
</tr>
<tr>
<td>F/2007</td>
<td>551</td>
<td>Design and Implementation of Operating Systems</td>
<td>3</td>
<td>70</td>
</tr>
<tr>
<td>F/2007</td>
<td>487</td>
<td>Software Engineering</td>
<td>3</td>
<td>40</td>
</tr>
<tr>
<td>S/2007</td>
<td>487</td>
<td>Software Engineering</td>
<td>3</td>
<td>38</td>
</tr>
<tr>
<td>F/2006</td>
<td>487</td>
<td>Software Engineering</td>
<td>6</td>
<td>70</td>
</tr>
</tbody>
</table>
13. Estimate the percentage of your time devoted to scholarly and/or research activities: 
\[ \frac{50}{\text{___}} \% \]. Please give a brief description of your major research and scholarly activities:

(1) Software composition and verification: Development of programming language constructs that enables programmer to develop reusable program modules not possible with existing general purpose programming languages. Programs written in this set of language constructs are compiled into objects that are particularly amenable for automatic assertion based verification.

(2) Multicast protocols: Efficient routing algorithms for multicast communications. Work includes network management of multicast communications (e.g. tunneling protocols).

(3) Automated software testing: tools and methods for unit and integration testing.

14. If you are a part time faculty member or a full-time faculty member without full-time commitment to the program, state what percentage of full-time you are assigned to the program: \[ \frac{100}{\text{___}} \% \].
1. Name, current academic rank, and tenure status:

<table>
<thead>
<tr>
<th>Name</th>
<th>Dennis A. Roberson</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rank</td>
<td>Research Professor</td>
</tr>
<tr>
<td>Tenure Status</td>
<td>Non-tenured</td>
</tr>
</tbody>
</table>

2. Date of original appointment to this faculty, followed by dates and ranks of advancement:

<table>
<thead>
<tr>
<th>Title</th>
<th>Dates Held</th>
</tr>
</thead>
<tbody>
<tr>
<td>Research Professor</td>
<td>June 2003 to present</td>
</tr>
</tbody>
</table>

3. Degrees with fields, institutions, and dates

<table>
<thead>
<tr>
<th>Degree</th>
<th>Field</th>
<th>Institution</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>MS</td>
<td>EE</td>
<td>Stanford University</td>
<td>1974</td>
</tr>
<tr>
<td>BS</td>
<td>Physics</td>
<td>Washington State University</td>
<td>1971</td>
</tr>
<tr>
<td>BS</td>
<td>EE</td>
<td>Washington State University</td>
<td>1971</td>
</tr>
</tbody>
</table>

4. If you do not have an advanced degree in the program area, describe any course work you may have taken, or other ways in which you have achieved competence in the program area; there is no necessity to repeat information here that is contained in later sections of this document.

Work Experience as described below. Various courses taken through companies over the years.

5. Conferences, workshops, and professional development programs in which you have participated in the last 5 years to improve teaching and professional competence in the program area:

<table>
<thead>
<tr>
<th>Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>Co-Chair, Third IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks, (DySPAN III), October 2008</td>
</tr>
<tr>
<td>Hosted annual NSF Wireless PI Meeting at IIT with roughly 100 of the nation’s top wireless researchers in attendance, July 2007</td>
</tr>
<tr>
<td>Technical Committee, First and Second IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks, (DySPAN II), April. 2007 and Nov. 2005</td>
</tr>
<tr>
<td>Technical Committee, First International Workshop on Technology and Policy for Accessing Spectrum (TAPAS), August 2006</td>
</tr>
</tbody>
</table>

6. Other related computing experience including teaching, industrial, governmental, etc. (where, when, description and scope of duties):
7. Consulting—list agencies and dates, and briefly describe each project:

<table>
<thead>
<tr>
<th>Dates</th>
<th>Agency</th>
<th>Project</th>
</tr>
</thead>
<tbody>
<tr>
<td>March 2008</td>
<td>National Academies / ONR</td>
<td>Countering the Threat of Improvised Explosive Devices</td>
</tr>
<tr>
<td>Feb 2008</td>
<td>Dept of Energy</td>
<td>Idaho National Labs Review</td>
</tr>
<tr>
<td>2002 - present</td>
<td>Malaysian Government</td>
<td>Multimedia Super Corridor International Advisory Panel (chaired by Prime Minister)</td>
</tr>
</tbody>
</table>

8. For the academic year in which the Self Study was written list your assigned duties other than for teaching, (committee membership, advising, etc.) with average hours per week. Indicate which, if any, carry extra compensation. If you are course coordinator for courses taught by other than full-time or part-time faculty, please indicate here which courses.

- Camras Scholars Program Committee and Research Pathway Director – 1 hour per week
- Illinois Tech Robotics Club – 2 hour per week
- University Corporate Relations – 20 hours per week
- Wireless Network & Communications Research Center – Co-Founder – 8 hours per week

9. For the four years preceding the Self Study, list all department, college, and/or university committees of which you are/were a member including year(s) served:

- Entrepreneurship Committee – 2006-2007
- Search Committee for Dean of Stuart School of Business – 2005-2006
- Undergraduate Business Academic Advisory Committee – 2003-2006
- Interprofessional Studies Committee – 2003- present

10. Principal publications during the last five years. Give in standard bibliographic format.


11. Other scholarly activity during the last 5 years: grants, sabbaticals, software development, etc.:

NSF Grants – 5 grants ($1.4M) to support research in Wireless Networking

12. Courses taught this and last academic year term-by-term. This year is the academic year in which this Self-Study report is prepared; the last year was the year prior to this. If you were on sabbatical leave, please enter the information for the year prior to the sabbatical. Please list each section of the same course separately.

<table>
<thead>
<tr>
<th>Term/year</th>
<th>Course number</th>
<th>Course title</th>
<th>Semester hrs</th>
<th>Class size</th>
</tr>
</thead>
<tbody>
<tr>
<td>S/2008</td>
<td>CS 595</td>
<td>Advanced Topics in Wireless Network Performance</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>F/2007</td>
<td>CS 495</td>
<td>Introduction to Wireless Network Performance</td>
<td>3</td>
<td>24</td>
</tr>
<tr>
<td>F/2006</td>
<td>BUS 100</td>
<td>Introduction to Business</td>
<td>3</td>
<td>36</td>
</tr>
<tr>
<td>S/2006</td>
<td>CS 595</td>
<td>Wireless Networks &amp; Performance</td>
<td>3</td>
<td>20</td>
</tr>
</tbody>
</table>

13. Estimate the percentage of your time devoted to scholarly and/or research activities: 50%. Please give a brief description of your major research and scholarly activities:
Wireless Networking Research focused on the areas of Interference and its impact on Network Performance and Spectrum Occupancy and Management

14. If you are a part time faculty member or a full-time faculty member without full-time commitment to the program, state what percentage of full-time you are assigned to the program: _10_% (not including research).
Senior Lecturer
1. Name, current academic rank, and tenure status:

<table>
<thead>
<tr>
<th>Name:</th>
<th>Matthew Bauer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rank:</td>
<td>Senior Lecturer</td>
</tr>
<tr>
<td>Tenure Status:</td>
<td>Not Tenured</td>
</tr>
</tbody>
</table>

2. Date of original appointment to this faculty, followed by dates and ranks of advancement:

<table>
<thead>
<tr>
<th>Title</th>
<th>Dates Held</th>
</tr>
</thead>
<tbody>
<tr>
<td>Instructor</td>
<td>May 1996-May 2000</td>
</tr>
<tr>
<td>Senior Lecturer</td>
<td>May 2000-present</td>
</tr>
</tbody>
</table>

3. Degrees with fields, institutions, and dates

<table>
<thead>
<tr>
<th>Degree</th>
<th>Field</th>
<th>Institution</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>BS</td>
<td>Math</td>
<td>IIT</td>
<td>6/86</td>
</tr>
<tr>
<td>MS</td>
<td>Math</td>
<td>IIT</td>
<td>12/87</td>
</tr>
<tr>
<td>PhD</td>
<td>CS</td>
<td>IIT</td>
<td>ABD</td>
</tr>
</tbody>
</table>

4. If you do not have an advanced degree in the program area, describe any coursework you may have taken, or other ways in which you have achieved competence in the program area; there is no necessity to repeat information here that is contained in later sections of this document.

All coursework completed for PhD in CS from IIT. ABD.

5. Conferences, workshops, and professional development programs in which you have participated in the last 5 years to improve teaching and professional competence in the program area:

IIT Workshop on Model Eliciting Activities  
CS Dept Workshop on Model Eliciting Activities  
ABET workshops on Assessment Process

6. Other related computing experience including teaching, industrial, governmental, etc. (where, when, description and scope of duties):

<table>
<thead>
<tr>
<th>Dates</th>
<th>Where</th>
<th>Duties</th>
</tr>
</thead>
<tbody>
<tr>
<td>6/88-8/96</td>
<td>Anderson Consulting</td>
<td>Computer Consulting</td>
</tr>
</tbody>
</table>

7. Consulting—list agencies and dates, and briefly describe each project:

NONE
8. For the academic year in which the Self Study was written list your assigned duties other than for teaching, (committee membership, advising, etc.) with average hours per week. Indicate which, if any, carry extra compensation. If you are course coordinator for courses taught by other than full-time or part time faculty, please indicate here which courses.

<table>
<thead>
<tr>
<th>IIT UG Studies Committee</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS Dept UG Studies Committee</td>
</tr>
<tr>
<td>IIT Director of Academic Advising (extra compensation)</td>
</tr>
</tbody>
</table>

9. For the four years preceding the Self Study, list all department, college, and/or university committees of which you are/were a member including year(s) served:

<table>
<thead>
<tr>
<th>IIT UG Studies Committee – every year</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS Dept UG Studies Committee – every year</td>
</tr>
</tbody>
</table>

10. Principal publications during the last five years. Give in standard bibliographic format.
NONE

11. Other scholarly activity during the last 5 years: grants, sabbaticals, software development, etc.:
NONE

12. Courses taught this and last academic year term-by-term. This year is the academic year in which this Self-Study report is prepared; the last year was the year prior to this. If you were on sabbatical leave, please enter the information for the year prior to the sabbatical. Please list each section of the same course separately.

<table>
<thead>
<tr>
<th>Term/year</th>
<th>Course number</th>
<th>Course title</th>
<th>Semester hrs</th>
<th>Class size</th>
</tr>
</thead>
<tbody>
<tr>
<td>2006F</td>
<td>CS100</td>
<td>Intro To Professions</td>
<td>2</td>
<td>43</td>
</tr>
<tr>
<td>2007S</td>
<td>CS201</td>
<td>Accelerated Intro to CS</td>
<td>4</td>
<td>25</td>
</tr>
<tr>
<td>2007S</td>
<td>CS430</td>
<td>Intro to Algorithms</td>
<td>3</td>
<td>43</td>
</tr>
<tr>
<td>2007F</td>
<td>CS116</td>
<td>Object-Oriented Prog. II</td>
<td>2</td>
<td>16</td>
</tr>
<tr>
<td>2007F</td>
<td>CS201</td>
<td>Accelerated Intro to CS</td>
<td>4</td>
<td>38</td>
</tr>
<tr>
<td>2008S</td>
<td>CS116</td>
<td>Object-Oriented Prog. II</td>
<td>2</td>
<td>36</td>
</tr>
</tbody>
</table>

13. Estimate the percentage of your time devoted to scholarly and/or research activities: 50%. Please give a brief description of your major research and scholarly activities:
Research in educational methods for CS0 and CS1 courses.

14. If you are a part time faculty member or a full-time faculty member without full-time commitment to the program, state what percentage of full-time you are assigned to the program: __100__%.
1. Name, current academic rank, and tenure status:

<table>
<thead>
<tr>
<th>Name</th>
<th>A. Mattox Beckman, Jr.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rank</td>
<td>Senior Lecturer</td>
</tr>
<tr>
<td>Tenure Status</td>
<td>Non-tenure track.</td>
</tr>
</tbody>
</table>

2. Date of original appointment to this faculty, followed by dates and ranks of advancement:

<table>
<thead>
<tr>
<th>Title</th>
<th>Dates Held</th>
</tr>
</thead>
<tbody>
<tr>
<td>Senior Lecturer</td>
<td>August 2003 – Present</td>
</tr>
<tr>
<td>Visiting Lecturer</td>
<td>August 2001 – August 2003</td>
</tr>
</tbody>
</table>

3. Degrees with fields, institutions, and dates

<table>
<thead>
<tr>
<th>Degree</th>
<th>Field</th>
<th>Institution</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>PhD</td>
<td>CS</td>
<td>University of Illinois at Urbana</td>
<td>12/2003</td>
</tr>
</tbody>
</table>

4. If you do not have an advanced degree in the program area, describe any course work you may have taken, or other ways in which you have achieved competence in the program area; there is no necessity to repeat information here that is contained in later sections of this document.

   NA

5. Conferences, workshops, and professional development programs in which you have participated in the last 5 years to improve teaching and professional competence in the program area:

   June 2007: Media Computation Workshop, Atlanta, GA
   June 2006: DisCoTec ’06 (Coordination language conference)
   April 2007: Assessment workshop (IIT)
   2006: Modeling expertise (IIT)
   2005: Lecture Preparation

6. Other related computing experience including teaching, industrial, governmental, etc. (where, when, description and scope of duties):

<table>
<thead>
<tr>
<th>Dates</th>
<th>Where</th>
<th>Duties</th>
</tr>
</thead>
</table>
7. Consulting—list agencies and dates, and briefly describe each project:

<table>
<thead>
<tr>
<th>Dates</th>
<th>Agency</th>
<th>Project</th>
</tr>
</thead>
<tbody>
<tr>
<td>2003—2006</td>
<td>Lawrence Cho IP</td>
<td>Patent application consulting</td>
</tr>
</tbody>
</table>

8. For the academic year in which the Self Study was written list your assigned duties other than for teaching, (committee membership, advising, etc.) with average hours per week. Indicate which, if any, carry extra compensation. If you are course coordinator for courses taught by other than full-time or part time faculty, please indicate here which courses.

Advising graduate students 0-10hpw, depending on time of year
Programming Language PhD Qualifying exam committee, 5 hours, twice a year

9. For the four years preceding the Self Study, list all department, college, and/or university committees of which you are/were a member including year(s) served:

Programming Language PhD Qualifying exam committee, 2003—present

10. Principal publications during the last five years. Give in standard bibliographic format.

System Imposed and Application Compliant Adaptations
24th International Conference on Distributed Computing Systems Workshops - W2: DARES (December 2003)

11. Other scholarly activity during the last 5 years: grants, sabbaticals, software development, etc.: 

12. Courses taught this and last academic year term-by-term. This year is the academic year in which this Self-Study report is prepared; the last year was the year prior to this. If you were on sabbatical leave, please enter the information for the year prior to the sabbatical. Please list each section of the same course separately.

<table>
<thead>
<tr>
<th>Term/year</th>
<th>Course number</th>
<th>Course title</th>
<th>Semester hrs</th>
<th>Class size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sp/08</td>
<td>CS 331</td>
<td>Data Structures</td>
<td>3</td>
<td>46</td>
</tr>
<tr>
<td>Sp/08</td>
<td>CS 440</td>
<td>Programming Languages</td>
<td>3</td>
<td>21</td>
</tr>
<tr>
<td>Fa/07</td>
<td>CS 105</td>
<td>Introduction to Programming</td>
<td>3</td>
<td>49</td>
</tr>
<tr>
<td>Fa/07</td>
<td>CS 331</td>
<td>Data Structures</td>
<td>3</td>
<td>51</td>
</tr>
<tr>
<td>Fa/07</td>
<td>CS 440</td>
<td>Programming Languages</td>
<td>3</td>
<td>24</td>
</tr>
<tr>
<td>Su/07</td>
<td>CS 440</td>
<td>Programming Languages</td>
<td>3</td>
<td>11</td>
</tr>
<tr>
<td>Semester</td>
<td>Course Code</td>
<td>Course Title</td>
<td>Credits</td>
<td>%</td>
</tr>
<tr>
<td>----------</td>
<td>-------------</td>
<td>----------------------</td>
<td>---------</td>
<td>---</td>
</tr>
<tr>
<td>Su/07</td>
<td>CS 536</td>
<td>Science of Programming</td>
<td>3</td>
<td>44</td>
</tr>
<tr>
<td>Sp/07</td>
<td>CS 331</td>
<td>Data Structures</td>
<td>3</td>
<td>31</td>
</tr>
<tr>
<td>Sp/07</td>
<td>CS 440</td>
<td>Programming Languages</td>
<td>3</td>
<td>24</td>
</tr>
<tr>
<td>Fa/06</td>
<td>CS 331</td>
<td>Data Structures</td>
<td>3</td>
<td>39</td>
</tr>
<tr>
<td>Fa/06</td>
<td>CS 440</td>
<td>Programming Languages</td>
<td>3</td>
<td>24</td>
</tr>
<tr>
<td>Fa/06</td>
<td>CS 541</td>
<td>Compilers</td>
<td>3</td>
<td>15</td>
</tr>
</tbody>
</table>

13. Estimate the percentage of your time devoted to scholarly and/or research activities: **15**%. Please give a brief description of your major research and scholarly activities:

Investigating the Actor Role Coordinator model with Shangping Ren. My contribution is to formalize the language constructs and their semantics.

14. If you are a part time faculty member or a full-time faculty member without full-time commitment to the program, state what percentage of full-time you are assigned to the program: _____%.
1. Name, current academic rank, and tenure status:

<table>
<thead>
<tr>
<th>Name:</th>
<th>James Sasaki</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rank:</td>
<td>Senior Lecturer</td>
</tr>
<tr>
<td>Tenure Status:</td>
<td>Not on tenure track</td>
</tr>
</tbody>
</table>

2. Date of original appointment to this faculty, followed by dates and ranks of advancement:

<table>
<thead>
<tr>
<th>Title</th>
<th>Dates Held</th>
</tr>
</thead>
<tbody>
<tr>
<td>Senior Lecturer</td>
<td>Aug 2003 - present</td>
</tr>
</tbody>
</table>

3. Degrees with fields, institutions, and dates

<table>
<thead>
<tr>
<th>Degree</th>
<th>Field</th>
<th>Institution</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>PhD</td>
<td>CS</td>
<td>Cornell University</td>
<td>1986</td>
</tr>
<tr>
<td>MS</td>
<td>CS</td>
<td>Cornell University</td>
<td>1984</td>
</tr>
<tr>
<td>BS</td>
<td>CS</td>
<td>Illinois Institute of Technology</td>
<td>1976</td>
</tr>
</tbody>
</table>

4. If you do not have an advanced degree in the program area, describe any course work you may have taken, or other ways in which you have achieved competence in the program area; there is no necessity to repeat information here that is contained in later sections of this document.

5. Conferences, workshops, and professional development programs in which you have participated in the last 5 years to improve teaching and professional competence in the program area:


6. Other related computing experience including teaching, industrial, governmental, etc. (where, when, description and scope of duties):

<table>
<thead>
<tr>
<th>Dates</th>
<th>Where</th>
<th>Duties</th>
</tr>
</thead>
</table>
7. Consulting—list agencies and dates, and briefly describe each project:

<table>
<thead>
<tr>
<th>Dates</th>
<th>Agency</th>
<th>Project</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

8. For the academic year in which the Self Study was written list your assigned duties other than for teaching, (committee membership, advising, etc.) with average hours per week. Indicate which, if any, carry extra compensation. If you are course coordinator for courses taught by other than full-time or part time faculty, please indicate here which courses.

Graduate committee, graduate admission, direct professional master's program, graduate advising for IIT India students, department web page development, liaison to university graduate admission, edit application-related materials, represent department at open houses and special registration. Apx 10 hrs/wk; receive summer support for these activities.

9. For the four years preceding the Self Study, list all department, college, and/or university committees of which you are/were a member including year(s) served:

   Academic years 2002–2006: Graduate committee, Graduate admission committee

10. Principal publications during the last five years. Give in standard bibliographic format.

    

11. Other scholarly activity during the last 5 years: grants, sabbaticals, software development, etc.:
12. Courses taught this and last academic year term-by-term. This year is the academic year in which this Self-Study report is prepared; the last year was the year prior to this. If you were on sabbatical leave, please enter the information for the year prior to the sabbatical. Please list each section of the same course separately.

<table>
<thead>
<tr>
<th>Term/year</th>
<th>Course number</th>
<th>Course title</th>
<th>Sem. hrs</th>
<th>Class size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fall'06</td>
<td>CS 330</td>
<td>Discrete Structures</td>
<td>3</td>
<td>32</td>
</tr>
<tr>
<td></td>
<td>CS 536</td>
<td>Science of Programming</td>
<td>3</td>
<td>051: 33, 092: 6, 251:22</td>
</tr>
<tr>
<td>Spring'07</td>
<td>CS 536</td>
<td>Science of Programming</td>
<td>3</td>
<td>051:20; 253: 5</td>
</tr>
<tr>
<td></td>
<td>CS 587</td>
<td>Software Project Management</td>
<td>3</td>
<td>9</td>
</tr>
<tr>
<td>Fall'07</td>
<td>CS 536</td>
<td>Science of Programming</td>
<td>3</td>
<td>001: 32, 251: 6</td>
</tr>
<tr>
<td></td>
<td>CS 587</td>
<td>Software Project Management</td>
<td>3</td>
<td>10</td>
</tr>
<tr>
<td>Spring'08</td>
<td>CS 201</td>
<td>Accelerated Intro to CS</td>
<td>4</td>
<td>28</td>
</tr>
<tr>
<td></td>
<td>CS 536</td>
<td>Science of Programming</td>
<td>3</td>
<td>051: 21, 536: 253</td>
</tr>
</tbody>
</table>

13. Estimate the percentage of your time devoted to scholarly and/or research activities: 0 %. Please give a brief description of your major research and scholarly activities:

14. If you are a part time faculty member or a full-time faculty member without full-time commitment to the program, state what percentage of full-time you are assigned to the program: _____%.
1. Name, current academic rank, and tenure status:

<table>
<thead>
<tr>
<th>Name</th>
<th>Marius D. Soneru</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rank:</td>
<td>Senior Lecturer</td>
</tr>
<tr>
<td>Tenure Status:</td>
<td>Non-tenure track</td>
</tr>
</tbody>
</table>

2. Date of original appointment to this faculty, followed by dates and ranks of advancement:

<table>
<thead>
<tr>
<th>Title</th>
<th>Dates Held</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lecturer, part-time</td>
<td>1981 - 1990</td>
</tr>
<tr>
<td>Adjunct Assistant/Associate Professor, part-time</td>
<td>1990 - 2001</td>
</tr>
<tr>
<td>Senior Lecturer, full time</td>
<td>2001 - present</td>
</tr>
</tbody>
</table>

3. Degrees with fields, institutions, and dates

<table>
<thead>
<tr>
<th>Degree</th>
<th>Field</th>
<th>Institution</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ph. D.</td>
<td>Computer Science</td>
<td>Illinois Institute of Technology</td>
<td>1981</td>
</tr>
<tr>
<td>M. S.</td>
<td>Computer Science</td>
<td>University of California at Los Angeles</td>
<td>1973</td>
</tr>
<tr>
<td>M.S. (5 year program)</td>
<td>E.E.</td>
<td>Polytechnic Institute of Bucharest</td>
<td>1965</td>
</tr>
</tbody>
</table>

4. If you do not have an advanced degree in the program area, describe any course work you may have taken, or other ways in which you have achieved competence in the program area; there is no necessity to repeat information here that is contained in later sections of this document.

N/A

5. Conferences, workshops, and professional development programs in which you have participated in the last 5 years to improve teaching and professional competence in the program area:

<table>
<thead>
<tr>
<th>Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>NXTcom07</td>
</tr>
<tr>
<td>Broadband Forum 06</td>
</tr>
<tr>
<td>Broadband Forum 05</td>
</tr>
<tr>
<td>National Communications Forum</td>
</tr>
<tr>
<td>International Conference on Multimedia Information Systems</td>
</tr>
</tbody>
</table>
6. Other related computing experience including teaching, industrial, governmental, etc. (where, when, description and scope of duties):

<table>
<thead>
<tr>
<th>Dates</th>
<th>Where</th>
<th>Duties</th>
</tr>
</thead>
<tbody>
<tr>
<td>1996-2001</td>
<td>Lucent Technologies, Bell Labs</td>
<td>Senior Technical Manager: Broadband systems Software Development</td>
</tr>
<tr>
<td>1990-1996</td>
<td>AT&amp;T Bell Labs</td>
<td>Technical Manager: Broadband systems Software Architecture</td>
</tr>
<tr>
<td>1981-1990</td>
<td>AT&amp;T Bell Labs</td>
<td>Supervisor: Broadband Switching Applied Research</td>
</tr>
<tr>
<td>1973-1981</td>
<td>AT&amp;T Bell Labs</td>
<td>Member/ Distinguished Member of Technical Staff: Next generation switching</td>
</tr>
</tbody>
</table>

7. Consulting — list agencies and dates, and briefly describe each project:

<table>
<thead>
<tr>
<th>Dates</th>
<th>Agency</th>
<th>Project</th>
</tr>
</thead>
</table>

8. For the academic year in which the Self Study was written list your assigned duties other than for teaching, (committee membership, advising, etc.) with average hours per week. Indicate which, if any, carry extra compensation. If you are course coordinator for courses taught by other than full-time or part-time faculty, please indicate here which courses.

M. S. Thesis Advisor: 2 hrs/wk

9. For the four years preceding the Self Study, list all department, college, and/or university committees of which you are/were a member including year(s) served:

10. Principal publications during the last five years. Give in standard bibliographic format.


11. Other scholarly activity during the last 5 years: grants, sabbaticals, software development, etc.:

12. Courses taught this and last academic year term-by-term. This year is the academic year in which this Self-Study report is prepared; the last year was the year prior to this. If you were on sabbatical leave, please enter the information for the year prior to the sabbatical. Please list each section of the same course separately.

<table>
<thead>
<tr>
<th>Term/year</th>
<th>Course number</th>
<th>Course title</th>
<th>Semester</th>
<th>Class size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fall 2006</td>
<td>CS455-001</td>
<td>Data Communications</td>
<td>3</td>
<td>26</td>
</tr>
<tr>
<td>Fall 2006</td>
<td>CS455-091</td>
<td>Data Communications</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>Fall 2006</td>
<td>CS455-251</td>
<td>Data Communications</td>
<td>3</td>
<td>20</td>
</tr>
<tr>
<td>Fall 2006</td>
<td>CS542-001</td>
<td>Computer Networks I</td>
<td>3</td>
<td>32</td>
</tr>
<tr>
<td>Fall 2006</td>
<td>CS548-051</td>
<td>Broadband Networks</td>
<td>3</td>
<td>30</td>
</tr>
<tr>
<td>Fall 2006</td>
<td>CS548-092</td>
<td>Broadband Networks</td>
<td>3</td>
<td>9</td>
</tr>
<tr>
<td>Fall 2006</td>
<td>CS548-251</td>
<td>Broadband Networks</td>
<td>3</td>
<td>19</td>
</tr>
<tr>
<td>Spring 2007</td>
<td>CS455-001</td>
<td>Data Communications</td>
<td>3</td>
<td>21</td>
</tr>
<tr>
<td>Spring 2007</td>
<td>CS455-091</td>
<td>Data Communications</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>Spring 2007</td>
<td>CS455-251</td>
<td>Data Communications</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>Spring 2007</td>
<td>CS548-051</td>
<td>Broadband Networks</td>
<td>3</td>
<td>38</td>
</tr>
<tr>
<td>Spring 2007</td>
<td>CS548-092</td>
<td>Broadband Networks</td>
<td>3</td>
<td>16</td>
</tr>
<tr>
<td>Spring 2007</td>
<td>CS548-251</td>
<td>Broadband Networks</td>
<td>3</td>
<td>25</td>
</tr>
<tr>
<td>Fall 2007</td>
<td>CS455-001</td>
<td>Data Communications</td>
<td>3</td>
<td>24</td>
</tr>
<tr>
<td>Fall 2007</td>
<td>CS455-091</td>
<td>Data Communications</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>Fall 2007</td>
<td>CS455-251</td>
<td>Data Communications</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>Fall 2007</td>
<td>CS548-001</td>
<td>Broadband Networks</td>
<td>3</td>
<td>28</td>
</tr>
<tr>
<td>Fall 2007</td>
<td>CS548-051</td>
<td>Broadband Networks</td>
<td>3</td>
<td>25</td>
</tr>
<tr>
<td>Fall 2007</td>
<td>CS548-251</td>
<td>Broadband Networks</td>
<td>3</td>
<td>16</td>
</tr>
<tr>
<td>Fall 2007</td>
<td>CS548-961</td>
<td>Broadband Networks</td>
<td>3</td>
<td>9</td>
</tr>
<tr>
<td>Spring 2008</td>
<td>CS455-001</td>
<td>Data Communications</td>
<td>3</td>
<td>21</td>
</tr>
<tr>
<td>Spring 2008</td>
<td>CS455-251</td>
<td>Data Communications</td>
<td>3</td>
<td>8</td>
</tr>
<tr>
<td>Spring 2008</td>
<td>CS548-051</td>
<td>Broadband Networks</td>
<td>3</td>
<td>24</td>
</tr>
<tr>
<td>Spring 2008</td>
<td>CS548-092</td>
<td>Broadband Networks</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>Spring 2008</td>
<td>CS548-251</td>
<td>Broadband Networks</td>
<td>3</td>
<td>9</td>
</tr>
</tbody>
</table>

13. Estimate the percentage of your time devoted to scholarly and/or research activities: 10%. Please give a brief description of your major research and scholarly activities:

IP Switching, Multiprotocol Label Switching (MPLS), ATM Congestion Control Methods, ATM Switching, Optical Networks, Broadband Access
14. If you are a part time faculty member or a full-time faculty member without full-time commitment to the program, state what percentage of full-time you are assigned to the program: _____%.
Instructor
1. Name, current academic rank, and tenure status:

<table>
<thead>
<tr>
<th>Name:</th>
<th>Omar Aldawud</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rank:</td>
<td></td>
</tr>
<tr>
<td>Tenure Status:</td>
<td></td>
</tr>
</tbody>
</table>

2. Date of original appointment to this faculty, followed by dates and ranks of advancement:

<table>
<thead>
<tr>
<th>Title</th>
<th>Dates Held</th>
</tr>
</thead>
<tbody>
<tr>
<td>Computer Faculty</td>
<td>Dec 2002</td>
</tr>
</tbody>
</table>

3. Degrees with fields, institutions, and dates

<table>
<thead>
<tr>
<th>Degree</th>
<th>Field</th>
<th>Institution</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>PhD</td>
<td>CS</td>
<td>Illinois Institute of Technology</td>
<td>Dec 2002</td>
</tr>
</tbody>
</table>

4. If you do not have an advanced degree in the program area, describe any course work you may have taken, or other ways in which you have achieved competence in the program area; there is no necessity to repeat information here that is contained in later sections of this document.

NA

5. Conferences, workshops, and professional development programs in which you have participated in the last 5 years to improve teaching and professional competence in the program area:

- Lead organizer AOM Workshop in conjunction with AOSD 2004 through 2008.
- http://www.aspect-modeling.org
- Models 2005, 2006 Program Committee
- AOSD 2009 Program Committee

6. Other related computing experience including teaching, industrial, governmental, etc. (where, when, description and scope of duties):

<table>
<thead>
<tr>
<th>Dates</th>
<th>Where</th>
<th>Duties</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oct 2006 –</td>
<td>Lucent Technologies</td>
<td>Software Architect, System Engineer and Software DDeveloper</td>
</tr>
<tr>
<td>Present</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

7. Consulting—list agencies and dates, and briefly describe each project:
<table>
<thead>
<tr>
<th>Dates</th>
<th>Agency</th>
<th>Project</th>
</tr>
</thead>
<tbody>
<tr>
<td>2005 –</td>
<td>Hostitwise.com</td>
<td>Ecommerce applications</td>
</tr>
<tr>
<td>2007</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

8. For the academic year in which the Self Study was written list your assigned duties other than for teaching, (committee membership, advising, etc.) with average hours per week. Indicate which, if any, carry extra compensation. If you are course coordinator for courses taught by other than full-time or part time faculty, please indicate here which courses.

9. For the four years preceding the Self Study, list all department, college, and/or university committees of which you are/were a member including year(s) served:

IIT / CS

10. Principal publications during the last five years. Give in standard bibliographic format.

- Lead organizer of the Aspect Oriented Modeling with UML Workshop held in Conjunction with Aspect Oriented Software Design (AOSD), March 2007, March 2006.
- Guest Reviewer DKE Journal
- Reviewer of special issue on "Auto-adaptive and Reconfigurable Systems" of the Wiley InterScience "Software, Practice and Experience" journal.
- Senior member Concurrent Programming Research Center.
- Tzilla Elrad, Omar Aldawud, Atef Bader. Aspect Oriented Modeling -


11. Other scholarly activity during the last 5 years: grants, sabbaticals, software development, etc.:

12. Courses taught this and last academic year term-by-term. This year is the academic year in which this Self-Study report is prepared; the last year was the year prior to this. If you were on sabbatical leave, please enter the information for the year prior to the sabbatical. Please list each section of the same course separately.

<table>
<thead>
<tr>
<th>Term/year</th>
<th>Course number</th>
<th>Course title</th>
<th>Semester hrs</th>
<th>Class size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fall/2007</td>
<td>CS 487</td>
<td>Software Engineering</td>
<td>3</td>
<td>59</td>
</tr>
<tr>
<td>Fall/2007</td>
<td>CS 445</td>
<td>OO Design and Programming</td>
<td>3</td>
<td>?</td>
</tr>
<tr>
<td>Spring 2008</td>
<td>CS 487</td>
<td>Software Engineering</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>Spring 2008</td>
<td>CSP 585</td>
<td>Design Patterns</td>
<td>3</td>
<td>33</td>
</tr>
</tbody>
</table>

13. Estimate the percentage of your time devoted to scholarly and/or research activities: 80 %. Please give a brief description of your major research and scholarly activities:

Advanced Object Oriented Design and Programming techniques such as Aspect Oriented Software Development. Also the applicability and impact of these techniques and methodologies on software engineering
14. If you are a part time faculty member or a full-time faculty member without full-time commitment to the program, state what percentage of full-time you are assigned to the program: ____%.
1. Name, current academic rank, and tenure status:

<table>
<thead>
<tr>
<th>Name:</th>
<th>Atef Bader</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rank:</td>
<td>Adjunct assistant professor</td>
</tr>
<tr>
<td>Tenure Status:</td>
<td></td>
</tr>
</tbody>
</table>

2. Date of original appointment to this faculty, followed by dates and ranks of advancement:

<table>
<thead>
<tr>
<th>Title</th>
<th>Dates Held</th>
</tr>
</thead>
<tbody>
<tr>
<td>part-time instructor</td>
<td>1997-1999</td>
</tr>
<tr>
<td>Adjunct assistant professor</td>
<td>2000-present</td>
</tr>
</tbody>
</table>

3. Degrees with fields, institutions, and dates

<table>
<thead>
<tr>
<th>Degree</th>
<th>Field</th>
<th>Institution</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>B.Sc.</td>
<td>CS</td>
<td>Yarmouk University</td>
<td>8/1988</td>
</tr>
<tr>
<td>M.Sc.</td>
<td>CS</td>
<td>Western Michigan University</td>
<td>4/1994</td>
</tr>
<tr>
<td>PhD</td>
<td>CS</td>
<td>Illinois Institute of Technology</td>
<td>5/1999</td>
</tr>
</tbody>
</table>

4. If you do not have an advanced degree in the program area, describe any coursework you may have taken, or other ways in which you have achieved competence in the program area; there is no necessity to repeat information here that is contained in later sections of this document.

5. Conferences, workshops, and professional development programs in which you have participated in the last 5 years to improve teaching and professional competence in the program area:

- Program committee member for The 5th ECOOP Workshop on Object-Orientation and Operating Systems, Malaga, Spain, 6/11/2002

6. Other related computing experience including teaching, industrial, governmental, etc. (where, when, description and scope of duties):
### Dates Where Duties

<table>
<thead>
<tr>
<th>Dates</th>
<th>Where</th>
<th>Duties</th>
</tr>
</thead>
</table>
| 01/01/97  | Alcatel-Lucent Technologies    | • System Engineer, System architect and lead engineer for the design and development of the ARTS (Automated Regression Testing System).  
• System lead for design and development for Network Management System/NBI  
• System lead for SIP/ISUP protocol test automation |

7. Consulting—list agencies and dates, and briefly describe each project:

<table>
<thead>
<tr>
<th>Dates</th>
<th>Agency</th>
<th>Project</th>
</tr>
</thead>
</table>

8. For the academic year in which the Self Study was written list your assigned duties other than for teaching, (committee membership, advising, etc.) with average hours per week. Indicate which, if any, carry extra compensation. If you are course coordinator for courses taught by other than full-time or part time faculty, please indicate here which courses.

9. For the four years preceding the Self Study, list all department, college, and/or university committees of which you are/were a member including year(s) served:

10. Principal publications during the last five years. Give in standard bibliographic format.


11. Other scholarly activity during the last 5 years: grants, sabbaticals, software development, etc.:

| Invited talk to Chicago Chapter of ACM – Aspect-Oriented Technology |

12. Courses taught this and last academic year term-by-term. This year is the academic year in which this Self-Study report is prepared; the last year was the year prior to this. If you were on sabbatical leave, please enter the information for the year prior to the sabbatical. Please list each section of the same course separately.

<table>
<thead>
<tr>
<th>Term/year</th>
<th>Course number</th>
<th>Course title</th>
<th>Semester hrs</th>
<th>Class size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spring/2008</td>
<td>CS -587-071</td>
<td>PROGRAM PROJECT MGT</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>Spring/2008</td>
<td>CS -587-251</td>
<td>PROGRAM PROJECT MGT</td>
<td>3</td>
<td>23</td>
</tr>
<tr>
<td>Spring/2008</td>
<td>CS -587-395</td>
<td>PROGRAM PROJECT MGT</td>
<td>3</td>
<td>26</td>
</tr>
<tr>
<td>Spring/2008</td>
<td>CSP -595-071</td>
<td>Unified Software Development Process</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>Spring/2008</td>
<td>CSP -595-251</td>
<td>Unified Software Development Process</td>
<td>3</td>
<td>10</td>
</tr>
<tr>
<td>Spring/2008</td>
<td>CSP -595-395</td>
<td>Unified Software Development Process</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>Fall/2007</td>
<td>CS -587-071</td>
<td>PROGRAM PROJECT MGT</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>Fall/2007</td>
<td>CS -587-251</td>
<td>PROGRAM PROJECT MGT</td>
<td>3</td>
<td>28</td>
</tr>
<tr>
<td>Fall/2007</td>
<td>CS -587-395</td>
<td>PROGRAM PROJECT MGT</td>
<td>3</td>
<td>19</td>
</tr>
<tr>
<td>Fall/2007</td>
<td>CSP -587-251</td>
<td>SOFTWARE QUALITY MANAGEMENT</td>
<td>3</td>
<td>15</td>
</tr>
<tr>
<td>Fall/2007</td>
<td>CSP -587-39</td>
<td>SOFTWARE QUALITY MANAGEMENT</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>Spring/2008</td>
<td>CS -587-</td>
<td>PROGRAM PROJECT MGT</td>
<td>3</td>
<td>19</td>
</tr>
</tbody>
</table>
13. Estimate the percentage of your time devoted to scholarly and/or research activities: 20%. Please give a brief description of your major research and scholarly activities:

Research related to Aspect-oriented technology and Adaptive software systems

14. If you are a part time faculty member or a full-time faculty member without full-time commitment to the program, state what percentage of full-time you are assigned to the program: 30%.
1. Name, current academic rank, and tenure status:

<table>
<thead>
<tr>
<th>Name:</th>
<th>Virgil Bistriceanu</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rank:</td>
<td>Instructor</td>
</tr>
<tr>
<td>Tenure Status:</td>
<td>Non-tenure track</td>
</tr>
</tbody>
</table>

2. Date of original appointment to this faculty, followed by dates and ranks of advancement:

<table>
<thead>
<tr>
<th>Title</th>
<th>Dates Held</th>
</tr>
</thead>
<tbody>
<tr>
<td>1994</td>
<td>Yearly contract renewal</td>
</tr>
</tbody>
</table>

3. Degrees with fields, institutions, and dates

<table>
<thead>
<tr>
<th>Degree</th>
<th>Field</th>
<th>Institution</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>MS</td>
<td>EE</td>
<td>“Polytechnic Institute”, Bucharest, Romania</td>
<td>1983</td>
</tr>
</tbody>
</table>

4. If you do not have an advanced degree in the program area, describe any course work you may have taken, or other ways in which you have achieved competence in the program area; there is no necessity to repeat information here that is contained in later sections of this document.

Course work complete for Ph.D. in Computer Science at the Illinois Institute of Technology. Qualifying examination passed.

5. Conferences, workshops, and professional development programs in which you have participated in the last 5 years to improve teaching and professional competence in the program area:

- RailsConf 2007 (http://conferences.oreillynet.com/rails2007/)

6. Other related computing experience including teaching, industrial, governmental, etc. (where, when, description and scope of duties):

<table>
<thead>
<tr>
<th>Dates</th>
<th>Where</th>
<th>Duties</th>
</tr>
</thead>
<tbody>
<tr>
<td>2002-2006</td>
<td>United Airlines</td>
<td>(i) Program Director for United’s new Internet Booking Engine (launched in December 2006): managed a $20 million budget and the activities of multiple</td>
</tr>
</tbody>
</table>
teams, including in-house and outsourced software development and testing.

(ii) Director, Software Development: managed the software development and testing teams for united.com, ameniti.com, mymileageplus.com, etc.


7. Consulting—list agencies and dates, and briefly describe each project:

<table>
<thead>
<tr>
<th>Dates</th>
<th>Agency</th>
<th>Project</th>
</tr>
</thead>
</table>

8. For the academic year in which the Self Study was written list your assigned duties other than for teaching, (committee membership, advising, etc.) with average hours per week. Indicate which, if any, carry extra compensation. If you are course coordinator for courses taught by other than full-time or part time faculty, please indicate here which courses.

9. For the four years preceding the Self Study, list all department, college, and/or university committees of which you are/were a member including year(s) served:

10. Principal publications during the last five years. Give in standard bibliographic format.

11. Other scholarly activity during the last 5 years: grants, sabbaticals, software development, etc.:
12. Courses taught this and last academic year term-by-term. This year is the academic year in which this Self-Study report is prepared; the last year was the year prior to this. If you were on sabbatical leave, please enter the information for the year prior to the sabbatical. Please list each section of the same course separately.

<table>
<thead>
<tr>
<th>Term/year</th>
<th>Course number</th>
<th>Course title</th>
<th>Semester hrs</th>
<th>Class size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spring/08</td>
<td>CS458</td>
<td>Information Security</td>
<td>3</td>
<td>14</td>
</tr>
<tr>
<td>Spring/08</td>
<td>CS445</td>
<td>Object-Oriented Design and Programming</td>
<td>3</td>
<td>28</td>
</tr>
<tr>
<td>Fall/07</td>
<td>CS470</td>
<td>Computer Architecture I</td>
<td>3</td>
<td>8</td>
</tr>
<tr>
<td>Fall/07</td>
<td>CS458</td>
<td>Information Security</td>
<td>3</td>
<td>17</td>
</tr>
<tr>
<td>Spring/07</td>
<td>CS445</td>
<td>Object-Oriented Design and Programming</td>
<td>3</td>
<td>11</td>
</tr>
<tr>
<td>Spring/07</td>
<td>CS458</td>
<td>Information Security</td>
<td>3</td>
<td>21</td>
</tr>
<tr>
<td>Fall/06</td>
<td>CS402</td>
<td>Intro to Advanced Studies II</td>
<td>3</td>
<td>8</td>
</tr>
<tr>
<td>Fall/06</td>
<td>CS470</td>
<td>Computer Architecture I</td>
<td>3</td>
<td>4</td>
</tr>
</tbody>
</table>

13. Estimate the percentage of your time devoted to scholarly and/or research activities: ___20__%. Please give a brief description of your major research and scholarly activities:

Software Engineering (large-scale agile software development)

14. If you are a part time faculty member or a full-time faculty member without full-time commitment to the program, state what percentage of full-time you are assigned to the program: _____%.
1. Name, current academic rank, and tenure status:

<table>
<thead>
<tr>
<th>Name:</th>
<th>Yonshik Choi</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rank:</td>
<td>Adjunct Faculty</td>
</tr>
<tr>
<td>Tenure Status:</td>
<td>NA</td>
</tr>
</tbody>
</table>

2. Date of original appointment to this faculty, followed by dates and ranks of advancement:

<table>
<thead>
<tr>
<th>Title</th>
<th>Dates Held</th>
</tr>
</thead>
<tbody>
<tr>
<td>Part Time Instructor</td>
<td>May 1998</td>
</tr>
</tbody>
</table>

3. Degrees with fields, institutions, and dates

<table>
<thead>
<tr>
<th>Degree</th>
<th>Field</th>
<th>Institution</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ph.D.</td>
<td>Computer Science</td>
<td>Illinois Institute of Technology</td>
<td>12/1998</td>
</tr>
<tr>
<td>MS</td>
<td>Computer Science</td>
<td>Illinois Institute of Technology</td>
<td>05/1989</td>
</tr>
<tr>
<td>BS</td>
<td>Industrial Engineering</td>
<td>Inha University</td>
<td>02/1987</td>
</tr>
</tbody>
</table>

4. If you do not have an advanced degree in the program area, describe any course work you may have taken, or other ways in which you have achieved competence in the program area; there is no necessity to repeat information here that is contained in later sections of this document.

5. Conferences, workshops, and professional development programs in which you have participated in the last 5 years to improve teaching and professional competence in the program area:

| Workshops for Faculty, Northwestern University, 2004. |
| Annual Professional Development Trainings, Lucent Technologies, etc. |

6. Other related computing experience including teaching, industrial, governmental, etc. (where, when, description and scope of duties):

<table>
<thead>
<tr>
<th>Dates</th>
<th>Where</th>
<th>Duties</th>
</tr>
</thead>
</table>

1998- current | Alcatel-Lucent (former Lucent Technologies) | Software development and engineering of telecommunication industry
2007 | Alcatel-Lucent | Taught technical classes of messaging network
2004 | Northwestern University | Taught Telecomm management and strategy
2005-2007 | Northwestern University | Taught Wireless Communications

7. Consulting—list agencies and dates, and briefly describe each project:

<table>
<thead>
<tr>
<th>Dates</th>
<th>Agency</th>
<th>Project</th>
</tr>
</thead>
<tbody>
<tr>
<td>1999</td>
<td>Lucent NKC</td>
<td>Transition Trouble Ticketing System (not as a faculty member, but as a Lucent staff)</td>
</tr>
<tr>
<td>2000- current</td>
<td>Alcatel-Lucent</td>
<td>Many Lucent internal consultation as a Lucent staff</td>
</tr>
</tbody>
</table>

8. For the academic year in which the Self Study was written list your assigned duties other than for teaching, (committee membership, advising, etc.) with average hours per week. Indicate which, if any, carry extra compensation. If you are course coordinator for courses taught by other than full-time or part-time faculty, please indicate here which courses.

9. For the four years preceding the Self Study, list all department, college, and/or university committees of which you are/were a member including year(s) served:

10. Principal publications during the last five years. Give in standard bibliographic format.


Examples of internal Lucent publications:


11. Other scholarly activity during the last 5 years: grants, sabbaticals, software development, etc.:


12. Courses taught this and last academic year term-by-term. This year is the academic year in which this Self-Study report is prepared; the last year was the year prior to this. If you were on sabbatical leave, please enter the information for the year prior to the sabbatical. Please list each section of the same course separately.

<table>
<thead>
<tr>
<th>Term/year</th>
<th>Course number</th>
<th>Course title</th>
<th>Semester hrs</th>
<th>Class size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spring/2008</td>
<td>CS401</td>
<td>Advance Study I</td>
<td>45</td>
<td>30</td>
</tr>
<tr>
<td>Spring/2008</td>
<td>CS542</td>
<td>Principle of Computer Networks</td>
<td>45</td>
<td>30</td>
</tr>
<tr>
<td>Fall/2007</td>
<td>CS401</td>
<td>Advance Study I</td>
<td>45</td>
<td>30</td>
</tr>
<tr>
<td>Fall/2007</td>
<td>CS402</td>
<td>Advance Study II</td>
<td>45</td>
<td>30</td>
</tr>
<tr>
<td>Summer/2007</td>
<td>CS455</td>
<td>Data Communications</td>
<td>45</td>
<td>30</td>
</tr>
<tr>
<td>Spring/2007</td>
<td>CS401</td>
<td>Advance Study I</td>
<td>45</td>
<td>30</td>
</tr>
<tr>
<td>Spring/2007</td>
<td>CS542</td>
<td>Principle of Computer Networks</td>
<td>45</td>
<td>30</td>
</tr>
</tbody>
</table>

13. Estimate the percentage of your time devoted to scholarly and/or research activities: ___10%. Please give a brief description of your major research and scholarly activities:


14. If you are a part time faculty member or a full-time faculty member without full-time commitment to the program, state what percentage of full-time you are assigned to the program: ___30%.
1. Name, current academic rank, and tenure status:

<table>
<thead>
<tr>
<th>Name</th>
<th>Jon Hanrath</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rank:</td>
<td>Senior Instructor</td>
</tr>
<tr>
<td>Tenure Status</td>
<td>N/A</td>
</tr>
</tbody>
</table>

2. Date of original appointment to this faculty, followed by dates and ranks of advancement:

<table>
<thead>
<tr>
<th>Title</th>
<th>Dates Held</th>
</tr>
</thead>
<tbody>
<tr>
<td>Senior Instructor</td>
<td>Fall 2002 - present</td>
</tr>
</tbody>
</table>

3. Degrees with fields, institutions, and dates

<table>
<thead>
<tr>
<th>Degree</th>
<th>Field</th>
<th>Institution</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>B.S.</td>
<td>Computer Science</td>
<td>University of Wisconsin – Madison</td>
<td>5/83</td>
</tr>
<tr>
<td>M.S.</td>
<td>Computer Science</td>
<td>University of Illinois – Urbana/Champaign</td>
<td>8/91</td>
</tr>
<tr>
<td>Ph. D.</td>
<td>Computer Science Education</td>
<td>Illinois Institute of Technology</td>
<td>Ongoing</td>
</tr>
</tbody>
</table>

4. If you do not have an advanced degree in the program area, describe any course work you may have taken, or other ways in which you have achieved competence in the program area; there is no necessity to repeat information here that is contained in later sections of this document.

NA

5. Conferences, workshops, and professional development programs in which you have participated in the last 5 years to improve teaching and professional competence in the program area:

Computer Science Education PhD program studies at IIT, including courses:

- MSED598 – Methods of College Teaching in Math/Science
- MSED552 – Assessment and Evaluation
- MSED603 – Qualitative Research and Practicum
- MSED604 – Qualitative Research and Practicum (Cont)

6. Other related computing experience including teaching, industrial, governmental, etc. (where, when, description and scope of duties):
Dates Where Duties
6/83 – 5/02 AT&T Bell Labs Computer Science Senior Instructor/Developer. Taught and developed software courses for internal and external students.

7. Consulting—list agencies and dates, and briefly describe each project:

<table>
<thead>
<tr>
<th>Dates</th>
<th>Agency</th>
<th>Project</th>
</tr>
</thead>
<tbody>
<tr>
<td>5/02 – present</td>
<td>Various software consulting companies</td>
<td>Consulted and taught software courses for various companies as a representative of Greenbriar Consulting and Training</td>
</tr>
</tbody>
</table>

8. For the academic year in which the Self Study was written list your assigned duties other than for teaching, (committee membership, advising, etc.) with average hours per week. Indicate which, if any, carry extra compensation. If you are course coordinator for courses taught by other than full-time or part time faculty, please indicate here which courses.

Judicial Board (1 hour a week).
Student Advising (2 hours a week).

9. For the four years preceding the Self Study, list all department, college, and/or university committees of which you are/were a member including year(s) served:

10. Principal publications during the last five years. Give in standard bibliographic format.

11. Other scholarly activity during the last 5 years: grants, sabbaticals, software development, etc.:

12. Courses taught this and last academic year term-by-term. This year is the academic year in which this Self-Study report is prepared; the last year was the year prior to this. If you were on sabbatical leave, please enter the information for the year prior to the sabbatical. Please list each section of the same course separately.

<table>
<thead>
<tr>
<th>Term/year</th>
<th>Course number</th>
<th>Course title</th>
<th>Semester hrs</th>
<th>Class size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fall/2007</td>
<td>cs105</td>
<td>Introduction to Programming using C++</td>
<td>135</td>
<td></td>
</tr>
<tr>
<td>Spring/2008</td>
<td>cs105</td>
<td>Introduction to Programming using C++</td>
<td>120</td>
<td></td>
</tr>
</tbody>
</table>
13. Estimate the percentage of your time devoted to scholarly and/or research activities: _10_. Please give a brief description of your major research and scholarly activities:

Research in computer science education as part of PhD program listed above.

14. If you are a part time faculty member or a full-time faculty member without full-time commitment to the program, state what percentage of full-time you are assigned to the program: _____%.
1. Name, current academic rank, and tenure status:

<table>
<thead>
<tr>
<th>Name</th>
<th>George Koutsogiannakis</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rank:</td>
<td>Instructor</td>
</tr>
<tr>
<td>Tenure Status:</td>
<td>None</td>
</tr>
</tbody>
</table>

2. Date of original appointment to this faculty, followed by dates and ranks of advancement:

<table>
<thead>
<tr>
<th>Title</th>
<th>Dates Held</th>
</tr>
</thead>
<tbody>
<tr>
<td>Instructor</td>
<td>2000-2008</td>
</tr>
</tbody>
</table>

3. Degrees with fields, institutions, and dates

<table>
<thead>
<tr>
<th>Degree</th>
<th>Field</th>
<th>Institution</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>B.S.</td>
<td>Physics</td>
<td>De Paul Universisty</td>
<td>1969</td>
</tr>
<tr>
<td>M.S.</td>
<td>Physics</td>
<td>De Paul University</td>
<td>1970</td>
</tr>
<tr>
<td>MBA</td>
<td>Finance</td>
<td>De Paul university</td>
<td>1990</td>
</tr>
<tr>
<td>M.S.</td>
<td>Computer Science</td>
<td>Illinois Institute Of Technology</td>
<td>1998</td>
</tr>
<tr>
<td>PhD</td>
<td>Computer Science</td>
<td>Illinois Institute Of Technology</td>
<td>Pending</td>
</tr>
</tbody>
</table>

4. If you do not have an advanced degree in the program area, describe any course work you may have taken, or other ways in which you have achieved competence in the program area; there is no necessity to repeat information here that is contained in later sections of this document.

NA

5. Conferences, workshops, and professional development programs in which you have participated in the last 5 years to improve teaching and professional competence in the program area:


6. Other related computing experience including teaching, industrial, governmental, etc. (where, when, description and scope of duties):

<table>
<thead>
<tr>
<th>Dates</th>
<th>Where</th>
<th>Duties</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dates</td>
<td>Agency</td>
<td>Project</td>
</tr>
<tr>
<td>-------------</td>
<td>-----------------------------</td>
<td>--------------------------------------------------------------------------</td>
</tr>
<tr>
<td>1999-2001</td>
<td>Computer Associates</td>
<td>Solving issues involving a number of products used by various customers.</td>
</tr>
<tr>
<td>2001-present</td>
<td>Neuroweb technologies</td>
<td>Self-Own company. Develop web sites for various customers.</td>
</tr>
</tbody>
</table>

8. For the academic year in which the Self Study was written list your assigned duties other than for teaching, (committee membership, advising, etc.) with average hours per week. Indicate which, if any, carry extra compensation. If you are course coordinator for courses taught by other than full-time or part time faculty, please indicate here which courses.

9. For the four years preceding the Self Study, list all department, college, and/or university committees of which you are/were a member including year(s) served:

10. Principal publications during the last five years. Give in standard bibliographic format.

- IEEE's International Performance of Computing and Communications Conference (IPCCC 2002- April 3, 2002-Phoenix, Arizona). Published and presented a paper on "Performance Studies on Java's Remote Method Invocation".

- ITPro Magazine (A IEEE publication of Computer Society). Published article: "Java Distributed Object models: An alternative to CORBA?", June 2002 issue.

- Third Workshop on Advances in Model Based Testing (A-MOST 2007), London England July 9, 2007--"Model-Based Test Prioritization Heuristic Methods and Their Evaluation", Korel, Koutsogiannakis

11. Other scholarly activity during the last 5 years: grants, sabbaticals, software development, etc.:
12. Courses taught this and last academic year term-by-term. This year is the academic year in which this Self-Study report is prepared; the last year was the year prior to this. If you were on sabbatical leave, please enter the information for the year prior to the sabbatical. Please list each section of the same course separately.

<table>
<thead>
<tr>
<th>Term/year</th>
<th>Course number</th>
<th>Course title</th>
<th>Semester hrs</th>
<th>Class size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spring 08</td>
<td>CS441</td>
<td>Current Topics In Programming Languages</td>
<td>3</td>
<td>26</td>
</tr>
<tr>
<td>Spring 08</td>
<td>CS402</td>
<td>Introduction to Advanced Studies II</td>
<td>3</td>
<td>7</td>
</tr>
<tr>
<td>Spring 08</td>
<td>CS115</td>
<td>Object Oriented Programming I</td>
<td>3</td>
<td>26</td>
</tr>
<tr>
<td>Fall 07</td>
<td>CS441</td>
<td>Current Topics In Programming Languages</td>
<td>3</td>
<td>21</td>
</tr>
<tr>
<td>Fall 07</td>
<td>CS115</td>
<td>Object Oriented Programming I</td>
<td>3</td>
<td>70</td>
</tr>
<tr>
<td>Summer 07</td>
<td>CS450</td>
<td>Operating Systems</td>
<td>3</td>
<td>22</td>
</tr>
<tr>
<td>Spring 07</td>
<td>CS441</td>
<td>Current Topics In Programming Languages</td>
<td>3</td>
<td>15</td>
</tr>
<tr>
<td>Spring 07</td>
<td>CS115</td>
<td>Object Oriented Programming I</td>
<td>3</td>
<td>15</td>
</tr>
<tr>
<td>Fall 06</td>
<td>CS441</td>
<td>Current Topics In Programming Languages</td>
<td>3</td>
<td>25</td>
</tr>
<tr>
<td>Fall 06</td>
<td>CS115</td>
<td>Object Oriented Programming I</td>
<td>3</td>
<td>70</td>
</tr>
</tbody>
</table>

13. Estimate the percentage of your time devoted to scholarly and/or research activities: ___50___%. Please give a brief description of your major research and scholarly activities:

Research on Model Based Test Prioritization Techniques.

14. If you are a part time faculty member or a full-time faculty member without full-time commitment to the program, state what percentage of full-time you are assigned to the program: _____%.
1. Name, current academic rank, and tenure status:

<table>
<thead>
<tr>
<th>Name:</th>
<th>Michael K. Saelee</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rank:</td>
<td>Full time instructor</td>
</tr>
<tr>
<td>Tenure Status:</td>
<td>Non-tenure / Non-tenure track</td>
</tr>
</tbody>
</table>

2. Date of original appointment to this faculty, followed by dates and ranks of advancement:

<table>
<thead>
<tr>
<th>Title</th>
<th>Dates Held</th>
</tr>
</thead>
<tbody>
<tr>
<td>Full-time instructor</td>
<td>9/2006-Present</td>
</tr>
</tbody>
</table>

3. Degrees with fields, institutions, and dates

<table>
<thead>
<tr>
<th>Degree</th>
<th>Field</th>
<th>Institution</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>BS</td>
<td>CS</td>
<td>Illinois Institute of Technology</td>
<td>1999</td>
</tr>
<tr>
<td>MS</td>
<td>CS</td>
<td>Illinois Institute of Technology</td>
<td>2001</td>
</tr>
</tbody>
</table>

4. If you do not have an advanced degree in the program area, describe any course work you may have taken, or other ways in which you have achieved competence in the program area; there is no necessity to repeat information here that is contained in later sections of this document.

NA

5. Conferences, workshops, and professional development programs in which you have participated in the last 5 years to improve teaching and professional competence in the program area:

CS 560: Computer Science in the Classroom

6. Other related computing experience including teaching, industrial, governmental, etc. (where, when, description and scope of duties):

<table>
<thead>
<tr>
<th>Dates</th>
<th>Where</th>
<th>Duties</th>
</tr>
</thead>
<tbody>
<tr>
<td>1/07-1/08</td>
<td>OptionsCity Software</td>
<td>Senior software engineer</td>
</tr>
</tbody>
</table>

7. Consulting—list agencies and dates, and briefly describe each project:

<table>
<thead>
<tr>
<th>Dates</th>
<th>Agency</th>
<th>Project</th>
</tr>
</thead>
<tbody>
<tr>
<td>3/08</td>
<td>Jones &amp; Bartlett</td>
<td>Programming textbook reviews</td>
</tr>
</tbody>
</table>
8. For the academic year in which the Self Study was written list your assigned duties other than for teaching, (committee membership, advising, etc.) with average hours per week. Indicate which, if any, carry extra compensation. If you are course coordinator for courses taught by other than full-time or part time faculty, please indicate here which courses.

<table>
<thead>
<tr>
<th>Advising</th>
</tr>
</thead>
<tbody>
<tr>
<td>Interviewing of prospective Camras scholars: (sporadic) 5 hrs</td>
</tr>
<tr>
<td>Course development: 10 hrs</td>
</tr>
</tbody>
</table>

9. For the four years preceding the Self Study, list all department, college, and/or university committees of which you are/were a member including year(s) served:

IPRO advisory committee, CAMRAS interview committee, ACM executive committee.

10. Principal publications during the last five years. Give in standard bibliographic format.

11. Other scholarly activity during the last 5 years: grants, sabbaticals, software development, etc.:

IPRO advising, software development and consulting.

12. Courses taught this and last academic year term-by-term. This year is the academic year in which this Self-Study report is prepared; the last year was the year prior to this. If you were on sabbatical leave, please enter the information for the year prior to the sabbatical. Please list each section of the same course separately.

<table>
<thead>
<tr>
<th>Term/year</th>
<th>Course number</th>
<th>Course title</th>
<th>Semester hrs</th>
<th>Class size</th>
</tr>
</thead>
<tbody>
<tr>
<td>F07</td>
<td>CS 105</td>
<td>Introduction to Computer Programming</td>
<td>3</td>
<td>55</td>
</tr>
<tr>
<td>F07</td>
<td>CS 351</td>
<td>Systems Programming</td>
<td>3</td>
<td>28</td>
</tr>
<tr>
<td>F07</td>
<td>CS 450</td>
<td>Operating Systems</td>
<td>3</td>
<td>41</td>
</tr>
<tr>
<td>S08</td>
<td>CS 351</td>
<td>Systems Programming</td>
<td>3</td>
<td>37</td>
</tr>
<tr>
<td>S08</td>
<td>CS 450</td>
<td>Operating Systems</td>
<td>3</td>
<td>47</td>
</tr>
</tbody>
</table>

13. Estimate the percentage of your time devoted to scholarly and/or research activities: 10%. Please give a brief description of your major research and scholarly activities:
Cross-lingual information retrieval literature review, related experiments.

14. If you are a part time faculty member or a full-time faculty member without full-time commitment to the program, state what percentage of full-time you are assigned to the program: _____%.
1. Name, current academic rank, and tenure status:

<table>
<thead>
<tr>
<th>Name</th>
<th>Alexander Manov</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rank</td>
<td>N/A, not teaching in Spring 2008</td>
</tr>
<tr>
<td>Tenure Status</td>
<td>N/A</td>
</tr>
</tbody>
</table>

2. Date of original appointment to this faculty, followed by dates and ranks of advancement:

<table>
<thead>
<tr>
<th>Title</th>
<th>Dates Held</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lecturer</td>
<td>1993-1995</td>
</tr>
<tr>
<td>Instructor</td>
<td>1995-2000</td>
</tr>
<tr>
<td>Senior Lecturer</td>
<td>2000-2004</td>
</tr>
</tbody>
</table>

3. Degrees with fields, institutions, and dates

<table>
<thead>
<tr>
<th>Degree</th>
<th>Field</th>
<th>Institution</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>PhD</td>
<td>CS</td>
<td>IIT, Chicago</td>
<td>May 2000</td>
</tr>
</tbody>
</table>

4. If you do not have an advanced degree in the program area, describe any course work you may have taken, or other ways in which you have achieved competence in the program area; there is no necessity to repeat information here that is contained in later sections of this document.

5. Conferences, workshops, and professional development programs in which you have participated in the last 5 years to improve teaching and professional competence in the program area:

6. Other related computing experience including teaching, industrial, governmental, etc. (where, when, description and scope of duties):

<table>
<thead>
<tr>
<th>Dates</th>
<th>Where</th>
<th>Duties</th>
</tr>
</thead>
</table>

7. Consulting—list agencies and dates, and briefly describe each project:

<table>
<thead>
<tr>
<th>Dates</th>
<th>Agency</th>
<th>Project</th>
</tr>
</thead>
</table>

VI - 143
8. For the academic year in which the Self Study was written list your assigned duties other than for teaching, (committee membership, advising, etc.) with average hours per week. Indicate which, if any, carry extra compensation. If you are course coordinator for courses taught by other than full-time or part time faculty, please indicate here which courses.

N/A

9. For the four years preceding the Self Study, list all department, college, and/or university committees of which you are/were a member including year(s) served:

N/A

10. Principal publications during the last five years. Give in standard bibliographic format.

11. Other scholarly activity during the last 5 years: grants, sabbaticals, software development, etc.:

12. Courses taught this and last academic year term-by-term. This year is the academic year in which this Self-Study report is prepared; the last year was the year prior to this. If you were on sabbatical leave, please enter the information for the year prior to the sabbatical. Please list each section of the same course separately.

<table>
<thead>
<tr>
<th>Term/year</th>
<th>Course number</th>
<th>Course title</th>
<th>Semester hrs</th>
<th>Class size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spring 08</td>
<td>Not teaching</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Fall 07</td>
<td>CSP527</td>
<td>Client/Server Application Development</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

13. Estimate the percentage of your time devoted to scholarly and/or research activities: _____%. Please give a brief description of your major research and scholarly activities:
14. If you are a part time faculty member or a full-time faculty member without full-time commitment to the program, state what percentage of full-time you are assigned to the program: ___0___%.
1. Name, current academic rank, and tenure status:

<table>
<thead>
<tr>
<th>Name:</th>
<th>Michael J. North</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rank:</td>
<td>Adjunct Professor of Computer Science</td>
</tr>
<tr>
<td>Tenure Status:</td>
<td>Non-tenured/Non-tenure track</td>
</tr>
</tbody>
</table>

2. Date of original appointment to this faculty, followed by dates and ranks of advancement:

<table>
<thead>
<tr>
<th>Title</th>
<th>Dates Held</th>
</tr>
</thead>
<tbody>
<tr>
<td>Adjunct Professor of Social Science</td>
<td>Spring 2006</td>
</tr>
<tr>
<td>Adjunct Professor of Computer Science</td>
<td>Spring 2008</td>
</tr>
</tbody>
</table>

3. Degrees with fields, institutions, and dates

<table>
<thead>
<tr>
<th>Degree</th>
<th>Field</th>
<th>Institution</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ph.D.</td>
<td>Computer Science</td>
<td>Illinois Institute of Technology</td>
<td>2005</td>
</tr>
<tr>
<td>MBA</td>
<td></td>
<td>Keller Graduate School of Management</td>
<td>1996</td>
</tr>
<tr>
<td>MS</td>
<td>Computer Systems Engineering</td>
<td>Illinois Institute of Technology</td>
<td>1995</td>
</tr>
<tr>
<td>MS</td>
<td>Computer Science</td>
<td>Governors State University</td>
<td>1994</td>
</tr>
<tr>
<td>BS</td>
<td>Computer Science</td>
<td>North Central College</td>
<td>1992</td>
</tr>
<tr>
<td>BA</td>
<td>Mathematics</td>
<td>North Central College</td>
<td>1992</td>
</tr>
<tr>
<td>AES</td>
<td></td>
<td>College of DuPage</td>
<td>1999</td>
</tr>
<tr>
<td>AGS</td>
<td></td>
<td>College of DuPage</td>
<td>1992</td>
</tr>
<tr>
<td>AS</td>
<td></td>
<td>College of DuPage</td>
<td>1991</td>
</tr>
<tr>
<td>AA</td>
<td></td>
<td>College of DuPage</td>
<td>1991</td>
</tr>
</tbody>
</table>

4. If you do not have an advanced degree in the program area, describe any course work you may have taken, or other ways in which you have achieved competence in the program area; there is no necessity to repeat information here that is contained in later sections of this document.
5. Conferences, workshops, and professional development programs in which you have participated in the last 5 years to improve teaching and professional competence in the program area:

<table>
<thead>
<tr>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Organizing committee member and Repast tutorial presenter for several international agent-based modeling and simulation in the social sciences conferences and workshops co-hosted by Argonne National Laboratory. Also, program committee co-chair for Agent 2004, Agent 2005, Agent 2006, and Agent 2007:</td>
</tr>
<tr>
<td>o Agent 2007 Conference on Complex Interaction and Social Emergence, Northwestern University, Evanston, IL USA (November 15-17, 2007).</td>
</tr>
<tr>
<td>o Agent 2006 Conference on Social Agents: Results and Prospects, University of Chicago, Chicago, IL USA (September 21-23, 2006).</td>
</tr>
<tr>
<td>o Agent 2004 Conference on Social Dynamics: Interaction, Reflexivity and Emergence, University of Chicago, Chicago, IL USA (October 7-9, 2004).</td>
</tr>
<tr>
<td>o Agent 2003 Conference on Challenges in Social Simulation, University of Chicago, Chicago, IL USA (October 3-4, 2003).</td>
</tr>
<tr>
<td>o Agent 2002 Workshop on Social Agents: Ecology, Exchange, and Evolution, University of Chicago, Chicago, IL USA (October 11-12, 2002).</td>
</tr>
<tr>
<td>o Agent 2000 Workshop on the Simulation of Social Agents: Architectures and Institutions, University of Chicago, Chicago, IL USA (October 6-7, 2000).</td>
</tr>
<tr>
<td>o Agent 1999 Workshop on Agent Simulation: Applications, Models, and Tools, University of Chicago, Chicago, IL USA (October 15-16, 1999).</td>
</tr>
<tr>
<td>Invited keynote speaker for the National Science Foundation Frontiers in Transport: Social Interactions workshop in Amsterdam, Netherlands (October 14-16, 2007).</td>
</tr>
<tr>
<td>Invited speaker for a Lund University doctoral course on complexity in operations and logistics management at Örenäs Castle, Glumslöv, Sweden (October 17, 2007).</td>
</tr>
<tr>
<td>Invited program committee member for the First World Congress on Social Simulation (WCSS), Kyoto, Japan (August 2006) and the Second World Congress on Social Simulation, Washington, DC, USA (August 2008).</td>
</tr>
</tbody>
</table>

6. Other related computing experience including teaching, industrial, governmental, etc. (where, when, description and scope of duties):

<table>
<thead>
<tr>
<th>Dates</th>
<th>Where</th>
<th>Duties</th>
</tr>
</thead>
<tbody>
<tr>
<td>1991</td>
<td>Argonne National</td>
<td>Currently Deputy Director, Center for Complex</td>
</tr>
</tbody>
</table>
Present Laboratory Adaptive Agent Systems Simulation (CAS²) within the Decision and Information Sciences Division. Responsibilities as the CAS² Deputy Director include program funding development and extensive project management, administrative management for six directly reporting employees, as well as developing detailed software designs, implementing extensive software, procuring software tools, and presenting the requirements and results to the research sponsors.

2005 – Present Joint Computation Institute, Argonne National Laboratory and the University of Chicago Currently a Senior Fellow.

7. Consulting—list agencies and dates, and briefly describe each project:

<table>
<thead>
<tr>
<th>Dates</th>
<th>Agency</th>
<th>Project</th>
</tr>
</thead>
</table>

8. For the academic year in which the Self Study was written list your assigned duties other than for teaching, (committee membership, advising, etc.) with average hours per week. Indicate which, if any, carry extra compensation. If you are course coordinator for courses taught by other than full-time or part time faculty, please indicate here which courses.

9. For the four years preceding the Self Study, list all department, college, and/or university committees of which you are/were a member including year(s) served:

10. Principal publications during the last five years. Give in standard bibliographic format.


Emonet, T., C.M. Macal, M.J. North, C.E. Wickersham, and P. Cluzel, “AgentCell: A


11. Other scholarly activity during the last 5 years: grants, sabbaticals, software development, etc.:

- Subcontract to the National Science Foundation Monitoring, Analysis, Diagnosis, and Control with Agent-Based Systems (MADCABS) Project (Award #0325378).
- Current leadership of the development of the free and open source Repast (http://repast.sourceforge.net/) agent-based modeling and simulation platform.
- Contributions to the development of the free and open source AgentCell (http://www.agentcell.org/) bacterial chemotaxis model.

12. Courses taught this and last academic year term-by-term. This year is the academic year in which this Self-Study report is prepared; the last year was the year prior to this. If you were on sabbatical leave, please enter the information for the year prior to the sabbatical. Please list each section of the same course separately.

<table>
<thead>
<tr>
<th>Term/year</th>
<th>Course number</th>
<th>Course title</th>
<th>Semester hrs</th>
<th>Class size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spring 2008</td>
<td>CS 495</td>
<td>Special Topics: Agent-based Modeling and Simulation</td>
<td>3</td>
<td>11</td>
</tr>
</tbody>
</table>

13. Estimate the percentage of your time devoted to scholarly and/or research activities: **100%**. Please give a brief description of your major research and scholarly activities:
I am a full time Computer Science researcher at Argonne National Laboratory with a focus on agent-based modeling and simulation (ABMS). I develop both ABMS platforms such as Repast (http://repast.sourceforge.net/) and ABMS models such as AgentCell (http://www.agentcell.org/). A substantial percentage of the research software that I develop is available on a free and open source basis. I often publish my research results in journals and peer reviewed conferences as indicted above.

14. If you are a part time faculty member or a full-time faculty member without full-time commitment to the program, state what percentage of full-time you are assigned to the program: 20%.
1. Name, current academic rank, and tenure status:

<table>
<thead>
<tr>
<th>Name:</th>
<th>Vida Winans</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rank:</td>
<td>Senior Instructor, Graduate Student Coordinator</td>
</tr>
<tr>
<td>Tenure Status:</td>
<td>N/A</td>
</tr>
</tbody>
</table>

2. Date of original appointment to this faculty, followed by dates and ranks of advancement:

<table>
<thead>
<tr>
<th>Title</th>
<th>Dates Held</th>
</tr>
</thead>
<tbody>
<tr>
<td>Instructor</td>
<td>8/2000-7/2002</td>
</tr>
<tr>
<td>Senior Instructor</td>
<td>8/2002-</td>
</tr>
</tbody>
</table>

3. Degrees with fields, institutions, and dates

<table>
<thead>
<tr>
<th>Degree</th>
<th>Field</th>
<th>Institution</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>B.A.</td>
<td>Chemistry</td>
<td>Cornell University</td>
<td>6/1975</td>
</tr>
<tr>
<td>M.S.</td>
<td>Computer Science</td>
<td>Illinois Institute of Technology</td>
<td>12/1990</td>
</tr>
</tbody>
</table>

4. If you do not have an advanced degree in the program area, describe any course work you may have taken, or other ways in which you have achieved competence in the program area; there is no necessity to repeat information here that is contained in later sections of this document.

5. Conferences, workshops, and professional development programs in which you have participated in the last 5 years to improve teaching and professional competence in the program area:

- CS 560 – CS in the Classroom, IIT: Fall 2003
- CS 561 – Computer & Curriculum, IIT: Spring 2004
- Modeling Eliciting Activity Workshop, IIT – Spring 2006

6. Other related computing experience including teaching, industrial, governmental, etc. (where, when, description and scope of duties):
<table>
<thead>
<tr>
<th>Dates</th>
<th>Where</th>
<th>Duties</th>
</tr>
</thead>
<tbody>
<tr>
<td>6/2005-</td>
<td>Argonne National Labs, Mathematics and</td>
<td>Part-time guest researcher – helped develop</td>
</tr>
<tr>
<td>8/2005</td>
<td>Computer Science Div.</td>
<td>bioinformatics programming tools</td>
</tr>
<tr>
<td>6/2006-</td>
<td>Argonne National Labs, Mathematics and</td>
<td>Part-time guest researcher – helped develop</td>
</tr>
<tr>
<td>8/2006</td>
<td>Computer Science Div.</td>
<td>bioinformatics programming tools</td>
</tr>
<tr>
<td>8/2007</td>
<td>Computer Science Div.</td>
<td>bioinformatics programming tools</td>
</tr>
</tbody>
</table>

7. Consulting—list agencies and dates, and briefly describe each project:

<table>
<thead>
<tr>
<th>Dates</th>
<th>Agency</th>
<th>Project</th>
</tr>
</thead>
</table>

8. For the academic year in which the Self Study was written list your assigned duties other than for teaching, (committee membership, advising, etc.) with average hours per week. Indicate which, if any, carry extra compensation. If you are course coordinator for courses taught by other than full-time or part time faculty, please indicate here which courses.

Graduate Student Coordinator – advising, prospective student inquiries, Graduate Admissions Open Houses & Orientations (5-6/year), scholarships/fellowships nominations, CS New Graduate Student Orientation, review of new student files for prerequisite courses, CS 201/CS 401 placement exam administration, CS 401/CS 402 proficiency exams administration; average hrs. per week: approx. 5-10 hrs./week, can be > 20 hrs./week during peak advising times including answering emails and phone advising for co-op students

CS Dept. Admissions committee – processing admissions, help maintain admissions data for dept., liaison between grad admissions and CS dept., respond to inquiries about admission decisions
Process non-Ph.D. applications during the summer: receive 1 month compensation for summer admissions work

CS Dept. – Graduate committee: NEA dept. accreditation course evaluations and course notebooks

9. For the four years preceding the Self Study, list all department, college, and/or university committees of which you are/were a member including year(s) served:

<table>
<thead>
<tr>
<th>Committees</th>
<th>Year(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS Dept. Admissions committee</td>
<td>2003-</td>
</tr>
<tr>
<td>CS Dept. Graduate committee</td>
<td>2005-</td>
</tr>
</tbody>
</table>

10. Principal publications during the last five years. Give in standard bibliographic format.
11. Other scholarly activity during the last 5 years: grants, sabbaticals, software development, etc.:

12. Courses taught this and last academic year term-by-term. This year is the academic year in which this Self-Study report is prepared; the last year was the year prior to this. If you were on sabbatical leave, please enter the information for the year prior to the sabbatical. Please list each section of the same course separately.

<table>
<thead>
<tr>
<th>Term/year</th>
<th>Course number</th>
<th>Course title</th>
<th>Semester hrs</th>
<th>Class size</th>
</tr>
</thead>
<tbody>
<tr>
<td>08S</td>
<td>CS 116-003, CS 116-004</td>
<td>Object-oriented Programming II</td>
<td>2</td>
<td>25</td>
</tr>
<tr>
<td>08S</td>
<td>CS 116-005, CS 116-006</td>
<td>Object-oriented Programming II</td>
<td>2</td>
<td>15</td>
</tr>
<tr>
<td>08S</td>
<td>IPRO 354</td>
<td>eMotion</td>
<td>3</td>
<td>9</td>
</tr>
<tr>
<td>07F</td>
<td>CS 115-003, CS 115-004</td>
<td>Object-oriented Programming I</td>
<td>2</td>
<td>40</td>
</tr>
<tr>
<td>07F</td>
<td>CS 115-007, CS 115-008</td>
<td>Object-oriented Programming I</td>
<td>2</td>
<td>40</td>
</tr>
<tr>
<td>07F</td>
<td>IPRO 354</td>
<td>eMotion</td>
<td>3</td>
<td>10</td>
</tr>
<tr>
<td>07S</td>
<td>CS 116-001, CS 116-002, CS 116-003</td>
<td>Object-oriented Programming II</td>
<td>2</td>
<td>50</td>
</tr>
<tr>
<td>07S</td>
<td>CS 116-004, CS 116-005</td>
<td>Object-oriented Programming II</td>
<td>2</td>
<td>35</td>
</tr>
<tr>
<td>06F</td>
<td>CS 201-001, CS 201-002</td>
<td>Accelerated Intro to CS</td>
<td>4</td>
<td>20</td>
</tr>
<tr>
<td>06F</td>
<td>CS 116-001</td>
<td>Object-oriented Programming II</td>
<td>2</td>
<td>12</td>
</tr>
</tbody>
</table>
13. Estimate the percentage of your time devoted to scholarly and/or research activities: __0___ %. Please give a brief description of your major research and scholarly activities:


14. If you are a part time faculty member or a full-time faculty member without full-time commitment to the program, state what percentage of full-time you are assigned to the program: ______%.
1. Name, current academic rank, and tenure status:

<table>
<thead>
<tr>
<th>Name:</th>
<th><strong>Hong Zhang</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Rank:</td>
<td>Research Assoc. Prof.</td>
</tr>
<tr>
<td>Tenure Status:</td>
<td>None</td>
</tr>
</tbody>
</table>

2. Date of original appointment to this faculty, followed by dates and ranks of advancement:

<table>
<thead>
<tr>
<th>Title</th>
<th>Dates Held</th>
</tr>
</thead>
<tbody>
<tr>
<td>Research Assoc. Prof</td>
<td>Aug., 2000</td>
</tr>
</tbody>
</table>

3. Degrees with fields, institutions, and dates

<table>
<thead>
<tr>
<th>Degree</th>
<th>Field</th>
<th>Institution</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ph.D</td>
<td>Applied</td>
<td>Michigan State Univ.</td>
<td>1989</td>
</tr>
</tbody>
</table>

4. If you do not have an advanced degree in the program area, describe any course work you may have taken, or other ways in which you have achieved competence in the program area; there is no necessity to repeat information here that is contained in later sections of this document.

5. Conferences, workshops, and professional development programs in which you have participated in the last 5 years to improve teaching and professional competence in the program area:

**Symposium Organization:**
“Parallel Implicit Approaches in Magnetic Fusion Applications” (with Lois Curfman McInnes and Thomas D. Rognlien), SIAM Conference on Parallel Processing for Scientific Computing, March 12-14, 2008, Atlanta, Georgia.

**Symposium Organization:**

6. Other related computing experience including teaching, industrial, governmental, etc. (where, when, description and scope of duties):

<table>
<thead>
<tr>
<th>Dates</th>
<th>Where</th>
<th>Duties</th>
</tr>
</thead>
</table>
7. Consulting—list agencies and dates, and briefly describe each project:

<table>
<thead>
<tr>
<th>Dates</th>
<th>Agency</th>
<th>Project</th>
</tr>
</thead>
</table>

8. For the academic year in which the Self Study was written list your assigned duties other than for teaching, (committee membership, advising, etc.) with average hours per week. Indicate which, if any, carry extra compensation. If you are course coordinator for courses taught by other than full-time or part time faculty, please indicate here which courses.

9. For the four years preceding the Self Study, list all department, college, and/or university committees of which you are/were a member including year(s) served:

10. Principal publications during the last five years. Give in standard bibliographic format.


11. Other scholarly activity during the last 5 years: grants, sabbaticals, software development, etc.:


12. Courses taught this and last academic year term-by-term. This year is the academic year in which this Self-Study report is prepared; the last year was the year prior to this. If you were on sabbatical leave, please enter the information for the year prior to the sabbatical. Please list each section of the same course separately.

<table>
<thead>
<tr>
<th>Term/year</th>
<th>Course number</th>
<th>Course title</th>
<th>Semester hrs</th>
<th>Class size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fall/07</td>
<td>CS21</td>
<td>Object Oriented Modeling and Design</td>
<td>3</td>
<td>66</td>
</tr>
<tr>
<td>Fall/06</td>
<td>CS521</td>
<td></td>
<td>3</td>
<td>40+</td>
</tr>
<tr>
<td>Fall/05</td>
<td>CS521</td>
<td></td>
<td>3</td>
<td>40+</td>
</tr>
<tr>
<td>Spring/06</td>
<td>CS590</td>
<td>Advanced Scientific Computing</td>
<td>3</td>
<td>10</td>
</tr>
<tr>
<td>Spring/03</td>
<td>CS401</td>
<td>Data Structure</td>
<td>3</td>
<td>40+</td>
</tr>
<tr>
<td>Fall/01</td>
<td>CS401</td>
<td></td>
<td>3</td>
<td>40+</td>
</tr>
<tr>
<td>Spring/01</td>
<td>CS401</td>
<td></td>
<td>3</td>
<td>40+</td>
</tr>
</tbody>
</table>

13. Estimate the percentage of your time devoted to scholarly and/or research activities: ____90__%. Please give a brief description of your major research and scholarly activities:


14. If you are a part time faculty member or a full-time faculty member without full-time commitment to the program, state what percentage of full-time you are assigned to the program: ____%. I normally work 10-20% on teaching at IIT, and 80-90% on research at Argonn National Lab.
C. Faculty Size

The purpose of this section is to determine whether you have a sufficient number of faculty members to provide overall continuity and stability for the program. The Faculty Profile table in 6.A relates to this concept.

1. Section 1 (Students) of the Self Study contains the course numbers of courses required for the major that are offered less frequently than once per year and those allowed for the major but not required, and explains how it is determined when they will be offered. Explain (if applicable) any difficulties you have offering required or optional courses frequently enough, particularly as they might be affected by faculty size.

We do not have any required CS courses that are offered less than once per year.

D. Faculty Workload

1. Describe the means for ensuring that all full-time faculty members have sufficient time for professional development and scholarly activities. For those faculty members having significant extra duties (e.g., large number of advisees, manage or maintain computing resources, director of undergraduate or graduate programs, etc.), explain how these components of the faculty workload are recognized.

The standard teaching load for tenure-track faculty is two courses per semester. For non-tenured, tenure-track faculty, to provide them with time to start a meaningful research program, only one course per semester is required. All tenure and tenure-track faculty have attended several top-tier international conferences in the past year, and most of the other full-time faculty members have attended at least one conference. Faculty can, and occasionally do, request and obtain the release of one course for research purposes.

Faculty can and often do obtain funding from external sources to cover released time for research. Full-time, non-tenure-track faculty members teach three courses per semester. We encourage these faculty members to submit research proposals that would provide funding to reduce their course load.

Faculty members with key advising loads or other administrative responsibilities are given sufficient time for professional development by reducing their required course load.

E. Program Development and Delivery

1. Describe the roles of the program’s faculty and other offices on the campus in creating, evaluating and modifying the program.
The Undergraduate Studies Committee works with course managers (all but two course managers are tenure-track faculty; Prof. Bauer is our key undergraduate advisor and Prof. Goharian has thrice been named the Department Teacher of the Year) and these course managers define objectives for each course. The faculty members who teach the courses are responsible for the creation of materials for the course and faculty can easily propose new courses. New courses are initially offered as CS 495 – a catchall number for a new, trial course. Once a course shows good enrollment as well as good student evaluations and the faculty member who taught the course agrees, the course is adopted into the general curriculum, the course is assigned a real number and brought into our curriculum. The Undergraduate Studies Committee, in concert with the faculty, is responsible for evaluating the consistency and quality of the courses and modifying the curriculum when necessary.

F. Course Oversight

1. Full-time faculty members have the responsibility for the consistency and quality of major courses. That means they must either teach all sections of a course or be responsible for coordinating the instruction of sections not taught by full-time faculty members. Describe how this oversight and coordination is performed.

Course Managers are assigned for each course, and, in conjunction with the Undergraduate Studies Committee, provide oversight and coordination with the faculty teaching each course.
7. Facilities

Criterion
Institutional facilities including the library, other electronic information retrieval systems, computer networks, classrooms, and offices are adequate to support the educational objectives and outcomes of the program. Computing resources are available, accessible, systematically maintained and upgraded, and otherwise adequately supported to enable students to achieve the program’s outcomes and to support faculty teaching needs and scholarly activities. Students and faculty receive appropriate guidance regarding the computing resources and laboratories available to the program.

A. Library Staffing

1. Assess the staffing of the library (or libraries) that serves the program, including both size and qualifications.

The library has one full-time librarian who is assigned as a subject specialist in Computer Science, along with other duties. This subject librarian acts as a liaison between the library and computer science faculty/staff/students and keeps communication open in regards to the library providing the resources that the department needs for research and scholarly activity. Research assistance is available from a professional, full-time librarian via phone, e-mail, instant messaging, or face-to-face.

B. Library Technical Collection

1. Assess the adequacy of the library’s technical collection relative to the needs of the program and the faculty. Describe and assess the adequacy of the process by which faculty may request the library to order books or subscriptions.

The library’s technical collection is quite adequate for the program at IIT. The Library subscribes to all the major databases and up-to-date collections both in print and electronic.

Key Full-text Databases in Computer Science:
   a. ACM Digital Library 1985-present
   b. IEEE Xplore 1950s-present
   c. SIAM Journals Vol. 1, no. 1-present
   d. Books24x7 Approx. 6,000 High-Quality e-Books

Other databases that contain full-text Computer Science content (trade journals, dissertations, etc.)
   a. Metapress (formerly SpringerLink and Kluwer journals)
   b. Wiley Interscience
   c. ScienceDirect
   d. Academic Search Premier
   e. ArticleFirst
The library offers a wide range of subscriptions and resources to the faculty, staff, and students in computer science. The library offers online access to all of the IEEE journals, standards, and technical reports from the 1950s or the first issue to the present. The library also subscribes to the ACM digital library, which includes all ACM publications, and the complete SIAM journal package, including their online archive, which provides access to every article published by SIAM. The library also provides access to a number of scholarly journals from Elsevier ScienceDirect, Metapress (Springer/Kluwer), and Wiley InterScience. Additionally, the library subscribes to Digital Dissertations giving access to Dissertation Abstracts, and the full-text of all Dissertations from 1997 to the present that are submitted to ProQuest (formerly UMI). In addition, the library provides access to full-text access to scholarly and trade journals through general databases such as Academic Search Premier, ArticleFirst, WilsonSelect, Business Source Premier, and Lexis/Nexis.

The library takes an approach to journal acquisitions that allows for access to the most resources possible on a limited budget. Many of the books for computer science classes are on reserve at the library, while articles, notes, past tests, etc. are on e-reserve, accessible to students at any time from anywhere.

To ensure the library always has the most up-to-date editions of key computer science titles, the library also subscribes to Books24x7, which is a collection of approximately 6,000 high quality, full-text e-books devoted solely to computer science and IT subjects from some of the top publishers in the world. It is now one of our most highly used databases. Access to e-books is made available in multiple ways; either through the library online catalog where there is a complete cataloging record and a web link to the book, or through the databases list and the CS user guide.

Faculty may contact the computer science librarian and share specific books from reading lists or books that are important to the research of the department. The computer science librarian can then make selections based on the budget that is available. The computer science librarian also selects books he or she thinks are appropriate or that have received widespread attention and sends these to the faculty liaison in computer science for approval. New books are featured in a special section after they arrive. The library currently has over 4,000 traditional printed books related to computer science.

C. Library Electronic Access
1. Assess the library’s systems for locating and obtaining electronic information.

The library has an online subject guide for computer science that points to high-quality databases and web resources and offers reference assistance. In addition, the library provides access to databases and electronic full-text journals for off-campus users through a proxy server that automatically generates accounts from enrollment records and makes all of the library’s digital resources available at any time from anywhere. The library includes computer science book records in the online catalog to make finding resources easier for the novice library user.

D. Classroom Equipment

1. Describe the equipment typically available in classrooms where you teach your courses. Assess its adequacy for the purpose.

Classrooms
IIT offers three levels of technology enhanced classrooms:

1. **Basic A/V classroom**, which is equipped with a network connection, a projector and screen, an ELMO and a VHS/DVD deck. All components are controlled through a single Crestron Control Panel on the instructor's desk.
2. **Distance Learning Classroom** has all the equipment of a basic A/V classroom, plus one or two video cameras, instructor and student microphones, plasma TV monitor, connections to broadcasting and digitizing devices for TV and/or Internet delivery. These classrooms also broadcast via television and the Internet.
3. **Video Conferencing Classroom**, which is similar to Distance Learning Classroom but also allows for real-time collaboration with a remote classroom location.

In addition, a **PC Classroom** is an OTS computer lab that is equipped with a PC and projector for the instructor and individual computers for each student. This arrangement provides students with a hands-on learning experience.

The following buildings are equipped with technology-enhanced learning classrooms:

Stuart Building:
- 8 basic A/V classrooms
- 8 distance learning classrooms (2 of which are videoconferencing classrooms)
- 4 PC classrooms

E1:
- 14 basic A/V classrooms
- 3 distance learning classrooms
- 1 PC classroom

See Appendix IV Facilities, Addendum D – spreadsheet
Alumni Hall:
- 2 basic A/V classrooms
- 1 PC classroom

Siegel Hall:
- 1 basic A/V classrooms
- 2 PC classrooms

E. Faculty Offices

1. Discuss and assess the adequacy of faculty offices.

Faculty offices are sufficient in that each faculty member has their own office with ample room for books, interaction with students, a workstation, network connection, bookshelves and other office equipment.

F. Computing Facilities

1. Describe the computing hardware, software and networks used for instruction. Specify any limitations that impact the quality of the educational experience.

Institutional and college computing facilities:

At the Illinois Institute of Technology, technology is an integrated part of the students’ education and development, as well as the faculty experience. The IIT community benefits from network infrastructure, computer classrooms and labs with current and relevant software, Blackboard online course management system, distance learning via Internet and IITV microwave channels, audio and visually equipped classrooms and administrative systems. The Office of Technology Services (OTS) operates these vital services and provides students with the technology that they need to access learning resources and thrive within their chosen fields of study.

In 2005, when IIT developed its 2010 Strategic Plan, extensive reports\(^2\) and student satisfaction surveys\(^3\) were used in order to gauge the effectiveness of existing technology resources. The Technology Satisfaction survey, an annual joint collaboration between OTS and TechNews, the IIT student newspaper, was used to determine student satisfaction. Through the reports and survey, it became clear that years of deferred technology maintenance had allowed infrastructure to age, which has impacted the university’s teaching, learning and research activities. To renew the university focus on technology, IIT established a Technology Platform Initiative to plan, prioritize and improve technology services. The Technology Platform

\(^2\) See Appendix IV Facilities, Addendum A – reports
\(^3\) See Appendix IV Facilities, Addendum B – survey
Committee developed a vision and implementation plan as well as defined technology standards the university will strive to meet in the next five years and beyond. These technology standards include network and Internet infrastructure, data security, and user interfaces.

The following is an outline of the technology and service enhancements deployed in the Main Campus and their impact on the engineering and computer science academic activities:

**Main Campus Infrastructure**

### Academic Buildings

From 2006 to 2008, OTS has upgraded technology at Engineering 1 (E1), Stuart, Metals, Siegel Hall, Life Sciences, Perlstein, and Crown Hall buildings. These recent upgrades include:

- Fiber connection into the buildings
- Fiber raiser between floors
- New teledata closets
- New network switches
- Infrastructure for Distance Learning
- A/V equipped classrooms/labs
- Full wireless coverage
- Replacement of CAT3 cables with CAT6 cables
- UPS in teledata

Engineering and computer science classroom and lab work activities are usually conducted in the following buildings: Stuart, E1, Alumni Hall, and Siegel Hall, each of which benefited from these upgrades⁴.

**OTS Computer Labs:**

OTS operates 12 labs in the Main Campus. The Stuart Building, E1, Alumni Hall and Siegel Hall computer labs were the focus of technology upgrades in 2006-2007 or are scheduled for upgrades within the next year. The Engineering and computer science student community usually use the labs in the following buildings:

The E1 building computer lab in room 029 has 21 workstations.
- Equipped with basic A/V System in Summer 2006.
- The 21 PCs that were refreshed in 2007 are due to be refreshed in Summer 2010.

The Stuart Building has four computer labs, with a total of 109 workstations.

**Stuart Lab 112J:**
- Equipped with basic A/V System in Summer 2006.
- The 46 PCs were refreshed in 2005 are due to be refreshed Summer 2008.

**Stuart Lab 112E:**
- The 22 PCs that were refreshed in 2006 are due to be refreshed in Summer 2009

---

⁴ See Appendix IV Facilities, Addendum C – spreadsheet
⁵ See Appendix IV Facilities, Addendum E – spreadsheet
⁶ See Appendix IV Facilities, Addendum F – learning materials
⁷ See Appendix IV Facilities, Addendum G – Blackboard learning materials
Stuart Lab 112F:
- The 22 PCs that were refreshed in 2006 are due to be refreshed in Summer 2009.

Stuart Lab 112X: (An open work area)
- The 19 PCs that were refreshed in 2006 are due to be refreshed in Summer 2009, with the addition of 4 new workstations.

Alumni Hall 218 has one computer lab with 29 workstations:
- The 29 PCs that were refreshed in 2007 are due to be refreshed Summer 2010.

Siegel Hall has two computer labs with 52 workstations:
Siegel Lab 237:
- The 31 PCs that were refreshed in 2006 are due to be refreshed Summer 2009.
Siegel Lab 236:
- The 21 PCs that were refreshed in 2007 are due to be refreshed Summer 2010.

The MTTC Night Owl Lab, opened in February 2006:
- The 50 laptops are due to be refreshed Summer 2008.

Software
OTS PC labs offer 81 current software titles that specifically address engineering students’ needs, and 49 titles that are geared toward Computer Science students. These titles are reviewed every semester by the IIT Software Committee, and are updated after thorough testing for compatibility with existing lab hardware/software5.

Distance Learning
IIT Online provides technology and procedural training6 for all new distance-learning faculty. The IIT Online Student Support Services provides assistance with delivery of course material and exams, exam/proctor coordination, delivering incoming material to instructors, and returning graded course material to students.

IIT Online offers 12 Chemical, Biological, Electrical and Computer Engineering Master’s degrees as distance learning programs:
1. Biomedical Imaging and Signals, Master (Internet and Televised)
2. Computer Engineering, M.S. (Internet and Televised)
3. Electrical and Computer Engineering, Master (Internet and Televised)
4. Electrical Engineering, M.S. (Internet and Televised)
5. Electricity Markets, Master (Internet and Televised)
6. Network Engineering, Master (Internet and Televised)
7. Power Engineering, Master (Internet and Televised)
8. Telecommunications and Software Engineering, Master (Internet and Televised)
9. VLSI and Microelectronics, Master (Internet and Televised)
10. Chemical Engineering, Master (Televised Only)
11. Environmental Engineering, Master (Televised Only)
12. Gas Engineering, Master (Internet Only)
IIT Online offers 15 graduate certificate programs, which provide students with post-baccalaureate knowledge of an area of specialization within chemical, biological, electrical or computer engineering. Students in these programs register as certificate students. Certificate programs typically require a set of three to four courses that must be completed in three years with a minimum GPA of 3.0/4.0. (Note: Some courses may have prerequisites.) Students who are admitted to master’s degree programs may apply coursework previously taken in a certificate program toward the requirements for the master’s degree.

The following Engineering certificate courses are offered as distance learning programs:

**Chemical and Biological Engineering Certificates**
- Bioengineering (Televised Only)
- Current Energy Issues (Internet Only)
- Hazardous Waste Engineering (Televised Only)
- Indoor Air Quality (Televised Only)
- Particle Processing (Televised Only)
- Pharmaceutical Processing (Televised Only)
- Polymer Synthesis and Characterization/Processing (Televised Only)
- Process Operations Management (Televised Only)
- Water and Wastewater Treatment (Televised Only)

**Electrical and Computer Engineering Certificates**
- Advanced Electronics (Internet and Televised)
- Computer Engineering (Internet and Televised)
- Control Systems (Internet and Televised)
- Power Engineering (Internet and Televised)
- Signal Processing (Televised Only)
- Wireless Communications (Internet and Televised)

**IIT Online distance learning programs offers three Computer Science Master’s degrees:**
1. Computer Science, Master (Internet and Televised) with specialization in:
   a. Computer Networking & Telecommunications,
   b. Information Systems
   c. Software Engineering
2. Computer Science, M.S. (Internet and Televised)
3. Computer Science, Master, Telecommunications and Software Engineering (Internet and Televised)

**Computer Science Certificates:**
1. Computer Networking and Telecommunications (Internet and Televised)
2. Information Systems (Internet and Televised)
3. Software Engineering (Internet and Televised)

**Blackboard**
Since 2003, the number of courses utilizing the Blackboard course management system has increased six fold. The Blackboard system hosts a website for every course offered at IIT and
serves as a portal to IIT Online streaming media, which can be accessed by students in both online and live course sections. Instructors post notes, lectures and assignments on the course page, which also features a discussion board and chat room.

In Fall 2008, 161 of the 579 (27.8%) Engineering courses use the Blackboard management system.

Each Fall, OTS conducts group Blackboard training\(^7\) for new professors. New professors arriving in Spring and Summer are offered either group or individual Blackboard training. Advanced Blackboard training sessions are also available for faculty currently using the system. IIT’s renewed focus on technology infrastructure and services have resulted in across the board improvements in service quality and user satisfaction demonstrated by service reports and the positive results of the Technology Satisfaction survey. The Armour College of Engineering and the College of Science and Letters’ students and faculty are a significant portion of the IIT community and thus are a primary beneficiary of those improvements.

Departmental computing facilities:

- 108SB – 30 dual-boot windows/linux PCs with software development environments in C++ and Java. Instructor workstation with Synchronize s/w, LCD, overhead
- omega.cs.iit.edu – student run linux server
- 2 Sun Fire V240 Machines with Solaris 10 OS. Each Machine with 2 CPUs - UltraSPARC-IIIi, CPU Speed 1503 MHz. Each Machine is equipped with 8 GB Memory Size, SCSI DISK—Size 146 GB.
- 1 Sun Ultra 5_10 with Solaris 8 with Memory 128 MB and 16 G of disk space. Hostname : cs450.cs.iit.edu
- 4 Ultra 5 Workstation Babbage cluster for cs546.
- 1 Pentium III Machine running Fedora Core release 5 for CS450, CS350, CS401, CS331, CS351 with 2 GB of memory, CPU Speed 500 Mhz with 2 Hard drives of 10 GB and 54.6 GB
- 1 Sun Fire V240. Babbage2.cs.iit.edu used by Faculty and students. Software provided is as follows: ORACLE, SPIM, MYSQL, APACHE, PHP, FORTE WORKSHOP COMPILERS for C++ and FORTRAN, MPI and babbage.cs.iit.edu for student class server with Software: OPNET, SPIM, GLOBUS, MYSQL, PHP, APACHE, NIS, and MPI. Babbage cluster nodes: sparc5.cs.iit.edu, sparc7.cs.iit.edu, sparc8.cs.iit.edu, sparc11.cs.iit.edu. Software on cluster is as follows: OPNET, SPIM, DINERO, MPI, WORKSHOP COMPILERS, and GLOBUS
- Software on all Solaris machines: gcc, flex, tar, make, autoconf, automake, sed, awk, shutils, binutils, ssh, vim, gtk+, pine, pico and so on

Other computing facilities:
• The Scalable Computing Software Laboratory has an 85-node Sun Microsystems ComputeFarm, a 17-node Dell PowerEdge Linux-based cluster, a 14-node IBM Linux-based cluster, a 12-node Cray XD1 supercomputer, a large Access Grid node, several multi-processor file servers, many high-end graphic workstations, and other advanced computing and communication facilities. The co-directors of the lab are Dr. Xian-He Sun and Dr. Zhiling Lan.

• The Information Retrieval Laboratory has 1 Sun Fire V880, 1 Sun Enterprise 450, 1 Netezza NPS 5200, 1 Compaq Proliant 6500, 1 Sun Ultra-10 Workstation, 4 Sun Ultra-5 Workstations, 1 MicroPC Server, 15 MicroPC Workstations and 2 Dell Inspiron Workstations. The director of the lab is Dr. Ophir Frieder, the IITRI chaired professor.

• The Lab for Visual Computing has 4 PCs running linux. The director of the lab is Dr. Gady Agam.

• The Network Systems Lab has 6 Sun Ultra-5 Workstations, 6 Sun Ultra-10 Workstations, 8 PCs running Windows, 2 PCs running Linux, all running OPNET, and a Cisco router. Dr. Cindy Hood is also in charge of an Access Grid node. Access Grid node has 3 Windows-based PCs.

• The lab for database and information systems has 8 late model Dell PCs with 3MHz dual core pentium CPUs and 4GB RAM each. It is led by Dr. Wai Gen Yee.

2. Describe the laboratory equipment planning, acquisition, and maintenance processes and their adequacy. Include discussion of these topics for university-wide computing resources available to all students (if used by your majors), your own laboratories and equipment (if applicable), and computing resources controlled by other departments and/or schools (if used by your majors). Discuss the adequacy and effectiveness of these processes and how they are assessed. Please attach documentation (e.g., inventories, equipment replacement plans, etc.) to this report.

The OTS-operated computer labs undergo frequent review and assessment to keep computing resources current and accessible. These computer labs are used for both academic courses and university-organized events. The lab computers are refreshed on a three-year cycle, to ensure that students have access to equipment that supports their academic goals. The lab instructional software is also reviewed every semester by the established IIT Software Committee, and is updated after thorough testing for compatibility with existing lab hardware and software.

All workstations and servers are managed and maintained by full-time professional staff in the Office of Technology Services. These staff members work with faculty and students to insure that the systems are reliable and the software is adequate for the coursework in the Computer Science department.

Adequacy of the computing facilities is assessed on a continuous basis, but usually coincides with the request to CS faculty each semester to provide any changes to the software and hardware for the next semester. In the majority the CS classes, the standard upgrade cycle is sufficient. In a few courses each year, specific hardware is upgraded and new software licenses are purchased as faculty introduce new courses. Very few student complaints are registered each semester to the system administrator, and students can voice concerns directly to the
university Chief Technology Officer if the situation is warranted.

3. What support personnel are available to install, maintain, and manage departmental/college hardware, software, and networks used for instruction in the program? Describe the adequacy and limitations of the level of support. Include discussions at the university, college and departmental levels as appropriate.

There is one full time support person, Upendra Gandhi, supporting CS Unix/Linux/Windows servers, department printers, and the SB 108 Lab. This person is employed by computing and network services and is tasked with supporting Computer Science requirements. This level of support is adequate.

G. Student Access

1. State the hours the various facilities are open. State whether students have access from dormitories or off campus by direct access, modem, etc., and describe this access quantitatively.

The network facilities at IIT were established to fulfill the educational mission of the university. OTS provides students, faculty, and staff access to an IIT Internet connection.

- Most main campus buildings have wired internet access. There is full wireless coverage in residence halls, MTCC, Galvin library, and all academic buildings. Partial wireless access is also available in some administrative buildings and public areas. View the current Main campus WiFi map at: http://ots.iit.edu/network/wireless.php

<table>
<thead>
<tr>
<th>Lab Location</th>
<th>Hours</th>
<th># of PCs</th>
<th>Phone</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stuart 112E, F, J, X</td>
<td>Monday - Friday, 8 am - Midnight</td>
<td>201</td>
<td>312-567-3792</td>
</tr>
<tr>
<td>3410 State 110</td>
<td>All Week, 9 am - 11 pm</td>
<td>30</td>
<td>312-567-8698</td>
</tr>
<tr>
<td>SR Crown 001</td>
<td>All Week, 9 am - 11 pm</td>
<td>34</td>
<td>312-567-8870</td>
</tr>
<tr>
<td>Siegel 236</td>
<td>Monday - Friday, 10 am - 9 pm</td>
<td>21</td>
<td>312-567-5805</td>
</tr>
<tr>
<td>Siegel 237</td>
<td>Monday - Friday, 8 am - 11 pm</td>
<td>31</td>
<td>312-567-3478</td>
</tr>
<tr>
<td>Alumni 218</td>
<td>Monday - Friday, 10 am - 9 pm</td>
<td>29</td>
<td>312-567-3572</td>
</tr>
<tr>
<td>E1 029</td>
<td>Monday - Friday, 9 am - 12 pm</td>
<td>21</td>
<td>312-567-3792</td>
</tr>
<tr>
<td>Building</td>
<td>Hours</td>
<td>Number</td>
<td>Phone</td>
</tr>
<tr>
<td>--------------------------</td>
<td>--------------------------------------------</td>
<td>--------</td>
<td>-----------------</td>
</tr>
<tr>
<td>Hermann Union Building</td>
<td>Monday - Friday, 8 am - 6 pm</td>
<td>20</td>
<td>312-567-5232</td>
</tr>
<tr>
<td>Hermann Union Building</td>
<td>Weekends, Closed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Public</td>
<td>Building Hours</td>
<td>6</td>
<td>312-567-5232</td>
</tr>
<tr>
<td>Rice 207, 210</td>
<td>Building Hours</td>
<td>42</td>
<td>630-682-6060</td>
</tr>
<tr>
<td>East Residents Hall 001</td>
<td>All Week, 6 am - 11 pm</td>
<td>15</td>
<td>312-808-6965</td>
</tr>
<tr>
<td>MTCC Owl Lab</td>
<td>All Week, 11 pm - 6 am</td>
<td>50</td>
<td>312-564-3375</td>
</tr>
<tr>
<td>MTCC Public</td>
<td>Building Hours</td>
<td>13</td>
<td>312-567-5232</td>
</tr>
</tbody>
</table>

All lab hour information is easily accessible for students and faculty members via the web at http://ots.iit.edu/computer_classrooms/

In addition, the students or faculty members can loan or check out some equipment from the lab, including a Digital Camcorders, cameras, laptops and projectors.

**H. Faculty Access**

1. Describe the computing facilities available to faculty for class preparation and for scholarly activities and research. Include specifics regarding resources in faculty offices.

- All faculty have PCs or workstations in their offices, access to computer classrooms, a high-speed black and white printer and a high-speed color printer in the department office, and access to laptops and projectors for use in classroom lectures or presentations.
- All faculty have opportunities to record their lectures using IIT/V course recording system and to tape presentations for scholarly conferences.
- IIT offers instructional design services, which entail creating materials to assist in faculty course development. These design services include adding media to classroom instruction and producing sound pedagogical content.
- All faculty have access to “Web for Faculty” for accessing their own class data, such as their class schedules and grading.
- Blackboard’s course management software is available for all faculty members. Accounts are formed upon creation of class assignment.
- Faculty members can request lab use for academic instruction through OTS on-line at http://www.enrollment.iit.edu/facstaff/clsroomres/clsreservform.html

OTS operates the IIT administrative systems, including WebForFaculty self-service systems that allow secure (SSL) access to student and staff information such as registration, grades, transcripts, class rosters, and employee benefits (http://my.iit.edu). IIT is in the middle of an
aggressive two-year process to upgrade to a unified digital campus (UDC) solution that will improve the efficiency of IIT’s business processes by providing users with access to a university-wide shared database and automating many processes that were previously done manually. The UDC includes "single sign-on" to personal information and resources through the secure web-based myIIT portal.
8. Support

Criterion

The institution’s support for the program and the financial resources available to the program are sufficient to attract and retain qualified faculty, administer the program effectively, acquire and maintain computing resources and laboratories, and otherwise provide an environment in which the program can achieve its educational objectives and outcomes. Support and resources are sufficient to provide assurance that the program will retain its strength throughout the period of accreditation.

A. Faculty Stability

1. Evidence of the long-term continuity and stability of a program is provided by its ability to both attract and retain high quality faculty. Describe how your program attracts and retains high quality faculty. Some topics the description might address are sabbatical and other leave programs, salaries, benefits, teaching loads, support for and recognition of scholarly activity (including financial support for attendance at professional meetings), departmental and institutional ambiance, etc.

The CS Department at IIT has been stable for over 30 years, hiring, promoting, and retaining quality faculty. The university (in the person of the dean and his superiors) and the department (in the person of the chair) strive to make the department a comfortable place to teach and do research. Salaries and teaching loads are quite competitive; TA support is sufficient. While the department has some funds for professional travel in cases of need, faculty are generally expected to cover the cost of such travel from external research grants. The university also returns a portion of grant overhead costs to cover such travel or other faculty needs.

The department is a friendly place with no internecine battles and a genuine spirit of camaraderie is pervasive. While our department is currently searching for a new permanent chair, the university administration has helped us work at attracting the most highly qualified candidates available. Our interim chair has, in the meantime, been working hard to develop connections within and outside IIT, fostering a sense of quality within the department, and making both research and teaching respected activities.

The University has a sabbatical policy in line with most universities (one semester at full pay or one year at half pay after 6 years) and gives leaves of absence without pay as appropriate for faculty development.

2. Give counts of the total number of full-time faculty and the number of resignations, retirements, and new hires for each of the last five years. Indicate whether there are significant problems attracting and retaining faculty, and if so, the causes.

<table>
<thead>
<tr>
<th>Year</th>
<th>Total Faculty</th>
<th>Resignations</th>
<th>Non-renewals</th>
<th>Retirements</th>
<th>New Hires</th>
</tr>
</thead>
<tbody>
<tr>
<td>2003-2004</td>
<td>29</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>2</td>
</tr>
</tbody>
</table>
2004-2005  29  0  0  0  0
2005-2006  29  1 (non-tenure track)  0  0  0
2006-2007  28  0  0  0  1
2007-2008  29  0  0  0  0

B. Faculty Professional Activities

1. Summarize the support mechanisms for professional activities of your faculty, such as attendance at meetings, research, etc. Highlight important faculty accomplishments that have resulted from this support.

As noted, departmental financial support is available for faculty attendance at research and other professional meetings, and such attendance is encouraged, both informally through departmental culture and through formal recognition in university publications and promotion standards.

Faculty accomplishments include the following:

**Major conferences organized or chaired (including as program committee chair):**

- AAIM 2007 - Chair
- ACM CIKM - Steering Committee
- ACM CIKM - General Chair
- ACM CIKM - Program Co-Chair
- ACM FOWANC 2008 - Chair
- ACM INFOSCALE - Steering Committee
- ACM MOBIHOC 2008
- ASI Workshop on Wireless Sensor Networks 2006 - Co-Organizer
- Biennial Israeli Symposium on Foundations of Artificial Intelligence (BISFAI)
- Chicago Colloquium on Digital Humanities and Computer Science - Founding Co-Chair
- IEEE High-Speed Local Networks Workshop 2004 - Program Chair
- IEEE International Conference on Information Technology (ITCC) 2004 - Conference Track Organizer
- IEEE International Conference on Software Maintenance 2004 - Program Co-chair
- IEEE Workshop on Cognitive Radio Networks 2007, 2008 - Program Chair
- Information and Knowledge Engineering (IKE) 2003 - Program Chair
- International Federation of Classification Societies (IFCS) 2004 - Conference Session Organizer
- MSN 2008
- NSF-sponsored workshop on Authorship Attribution Research - Organizer
- NSF/DIMACS Workshop on Optical Networks 1998 - Co-Organizer
- SPIE Optical Networking and Communications Conference 2001
Society fellowships:
2 ACM Fellows
2 IEEE Fellows
1 AAAS Fellow

Journals edited:
Ad Hoc & Sensor Wireless Networks: An International Journal
DKE Journal
EUROMICRO Journal.
IEEE Symposia on New Frontiers in Dynamic Spectrum Access Networks
International Journal of Innovation Science
International Journal of Network Management
Journal of the American Society for Information Science & Technology
Journal of Supercomputing
Journal of Systems Architecture

Journal special issues edited:
ACM MONET on non-cooperative computing in wireless networks, 2004-2005 - Guest Editor of Special Issue
Empirical Software Engineering Journal 2006 - Co-editor of the Special Issue
IEEE JSAC Special Issue on Non-Cooperative Issues in Distributed Computing and Networking, 2005-2006 - Guest Editor
International Journal of Wireless and Mobile Computing (IJWMC), 2006 - Co-Guest Editor of Special Issue
Journal of Automated Software Engineering (in progress) - Co-editor of the Special Issue
Journal of the American Society for Information Science on Computational Stylistics - Editor of Special Issue
Wiley InterScience "Software, Practice and Experience" journal, "Auto-adaptive and Reconfigurable Systems" - Guest Editor of Special Issue

Books written or edited: 18

Journal Papers published in the last 5 years: Over 125

C. Administration Effectiveness

1. Describe the effectiveness of the administration of the program.
Alumni surveys have repeatedly revealed that the department has successfully met its goals. Additionally, students of the last five years have obtained positions with top high technology companies as well as admission to top graduate programs. This certainly suggests that the administration of the program is reasonably effective.

The department dedicates significant resources to administration. The department chair, the deputy department chair and the Undergraduate Program Director are all extremely focused on administration. Additionally, the Undergraduate Studies Committee meets regularly to administer, examine, and revise the undergraduate program.

D. Adequacy of Resources

1. Describe the adequacy of the resources available to the program, including those to acquire and maintain laboratory facilities, relative to the ability of the program to achieve its educational objectives and outcomes. Include information on how the institution determines the adequacy of these resources.

**Classrooms:** Most classrooms used by the program have been recently renovated to high technical and aesthetic standards. The classrooms have whiteboards together with integrated document/computer display facilities, which can project on an in-room screen as well as to television monitors in remote locations; classrooms have cameras to film the lecturers as well for this purpose. Video is also often made available for students on the internet. Most classrooms comfortably seat 25 students.

**Laboratory Facilities:** Computing and Network Services has installed some of the most current software and hardware in its computing labs across campus. Faculty who teach a course are asked precisely what they need with regards to software and hardware required by a course the semester before they teach the course. In most cases, the faculty requests are granted. A detailed list of facilities available to faculty is found in Section 7.

The committee recommendations and meetings determine the adequacy of the resources with students, faculty and staff. IIT has formal Information Technology Advisory Council that acts as a focal point for IT-related committees throughout the university. New equipment is specified and purchased based on these recommendations. Workstations and servers are replaced on an ongoing basis, but the replacement is usually driven by changes in the software required by academic departments. As a result, workstations are replaced every 18 to 36 months.

**Library Facilities:** Illinois Institute of Technology meets its community’s information needs through six libraries. These libraries are located on the university’s four campuses. The Paul V. Galvin Library serves as the main library of Illinois Institute of Technology. Each of the libraries covers one or more areas of knowledge, carries an array of special material, and offers multiple services related to the disciplines it covers. The library’s physical presence is significantly enhanced by its online presence,
providing remote, uninterrupted access to many of our services and collections, including electronic journal subscriptions. Library personnel actively seek feedback and suggestions from faculty and students. The Dean of Libraries reports to the Academic Vice President/Provost.

E. Continuity of Institutional Support

1. Discuss and show evidence of continuity of institutional support for the program in the past, and problems that have existed or are anticipated in this area, if any.

The department has been educating students in computer science for over a quarter of a century. Certainly, there have been fluctuations of interest in Computer Science over the years. Since we provide a solid stream of students and course offerings, we receive significant support from the administration. As evidence, we have hired three new faculty in the past five years and have completed a $200,000 renovation of our teaching lab (108 Stuart) in Summer 2006, as well as other important renovations to our departmental facilities.

Classroom upgrades were completed in the following buildings
- Engineering 1, Wishnick Hall, Siegel Hall (summer 2006)
- Stuart Building (Summer 2007) – the majority of CS courses are taught in this building
- Life Sciences (Summer 2008)
APPENDIX I – Institutional Summary

The institution may employ any means it chooses to represent itself to ABET and the visiting team. Consequently, the references to specific tables in the following are for guidance only. The information may be presented in any manner the institution chooses.

The Institution

Illinois Institute of Technology
Chief Executive Officer: John L. Anderson, President

Type of Control

Illinois Institute of Technology is an independent non-sectarian, co-educational, urban university. It is governed by a board of trustees drawn from diverse groups representing the public interest.

History of Institution

Armour Institute opened in 1893; the institute offered professional courses in engineering, chemistry, architecture and library science. IIT was created in 1940 by the merger of Armour Institute with Lewis Institute (est. 1895), a West Side Chicago college that offered liberal arts as well as science and engineering courses. The Institute of Design, founded in 1937, merged with IIT in 1949.

In 1969, IIT became one of the few technology-based universities with a law school when the Chicago Kent College of Law, founded in 1887, became an integral part of the university. Stuart School of Business was added in 1969, with a gift from the estate of Lewis Institute alumnus and Chicago financier Harold Leonard Stuart. The school became the Stuart School of Business in 1999. Midwest College of Engineering, founded in 1967, joined the university in 1986, forming the nucleus for IIT's west suburban campus.

Today, IIT is a private, Ph.D.-granting university with programs in engineering, science, psychology, architecture, business, design and law. It is one of the 16 institutions that comprise the Association of Independent Technological Universities (AITU).

Student Body

Please see included tables. The IIT student body is exceptionally diverse; students are drawn from all 50 states of the USA, and from over 90 nations.
Admissions Process

Admission decisions are based on academic performance, standardized test scores, teacher/counselor recommendations and evidence of promise to succeed, which includes co-curricular activities, interests and hobbies, and personal maturity.

Students must have attended an accredited high school (although we do accept home schooled students) and have completed a minimum of 16 units of high school work and a minimum of 3½ units of mathematics that must include 2 units of algebra through pre-calculus, 1 unit of geometry and ½ unit of trigonometry. Calculus is strongly recommended but not required. Additionally, students must have completed 2 units of laboratory science (preferably physics and chemistry). Students are encouraged to take an additional laboratory science. Additional requirements include 4 units of English, and 2 units of History or Social Studies.

It is expected that students select a rigorous high school program that includes AP, IB or honors courses when they are available at the student's school. Students are encouraged to take college courses to supplement their education while they are enrolled in high school.

Students with unweighted grade point averages greater or equal to 3.0 and ACT test scores greater or equal to 24 math and 24 composite, or SAT scores greater or equal to 1150 may be admitted without a faculty committee review. Students who fall below these floors are generally denied admission, but may be, on an individual basis, selected for admission by a faculty review committee.

IIT recognizes and grants credit for students who have satisfactory scores for Advanced Placement or International Baccalaureate examinations. IIT also will grant transfer credit for college course work taken while a student was in high school provided a grade of “C” or above was earned.

IIT does not have an “upper division” per se. Students admitted as “Undeclared Engineering” are subject to the same requirements as all other admits. They are expected to declare a major by the end of the first year of study.

The Office of Educational Services is responsible for verifying all courses transferred from other colleges. Transfer applicants must be in good academic standing at their previous colleges to be considered for admission to IIT. Applicants with less than 30 hours of transferable college course work must submit high school transcripts and SAT or ACT scores as part of their application. Admission is based upon a cumulative GPA and individual grades in all classes that apply to the selected major. A minimum cumulative GPA of 3.0 is expected for transfer consideration. However, a transfer applicant who has special circumstances will be reviewed by a faculty committee.
Transfer credit is granted only for courses completed at schools listed in *Transfer Credit Practices of Designated Educational Institutions, American Association of Collegiate Registrars and Admissions Officers*. For engineering students, transfer credit for the equivalent of engineering and professional electives is given only for courses completed at schools accredited by the EAC of ABET.

Transfer credit is granted on a course equivalency basis, i.e., the nature, content, level and prerequisites of the course must be comparable to those offered at IIT. Students may transfer a maximum of 68 applicable credits from a 2-year college. Transfer students must complete their last 45 credits at IIT with at least 50% of the course work at the 300 and 400 level in their major discipline. Transfer credit will be accepted for courses completed with the equivalent of a grade of “C” or better.

Joint programs with specific articulation agreements have been established with Benedictine University, DePaul University, Dominican University, University of St. Francis, Elmhurst College, and Wheaton College. Depending on the specific partner institution, students may receive a degree in Aerospace Engineering, Architectural Engineering, Chemical Engineering, Civil Engineering, Computer Engineering, Electrical Engineering, or Mechanical Engineering and a Bachelor’s degree in an approved discipline from their host school. Students are considered full-time at their host institution while completing requirements for both degrees. Admission into the Joint Program at another institution does not guarantee admission to IIT. Students must meet IIT admission requirements. Grades of “D” are acceptable for transfer credit for general education courses only. Programs of study have been produced for all engineering curricula available at each specific partner institution.

**Regional or Institutional Accreditation**

Illinois Institute of Technology has had continuous accreditation from the North Central Association of Colleges and Schools since 1941; the last accreditation visit was in 2006.

**Personnel and Policies**

(a) Promotion and tenure policies

Tenure track and tenured ranks are: assistant professor, associate professor, and professor. Decisions on promotion and tenure are, by authority of the Board of Trustees, vested in the President of the university. For each candidate, the Provost is expected to make recommendations to the President based on consideration of university needs, plans, and resources, and on the recommendations submitted by the following faculty committees and individuals:

1. the Academic Unit Committee on Promotion and Tenure (AUCOPT);
2. the Campus Committee on Promotion and Tenure (CAMCOPT);
3. the University Committee on Promotion and Tenure (UCOPT); and
4. the head of the academic unit and, in the case of a college with departments, the dean of the candidate’s department.

The recommendations of the faculty committees as to any candidate are the result of the consideration of the portfolio of the candidate and any additional information or recommendations provided at the request of the committees by appropriate persons, including the candidate, professional peers from outside IIT, fellow faculty members, the Provost, the academic unit head, and students.

Evaluation of candidates for tenure appointments and for promotions to the rank of professor are based on clearly defined standards of academic quality. Inasmuch as there may be significant differences in the spirit and traditions of the individual disciplines comprising IIT, standards may vary from one profession to another. While each academic unit is expected to formulate its own standards and guidelines for the evaluation of its faculty, the following criteria are common to all academic units: performance in teaching, advising and the promotion of student learning; scholarly activities appropriate to the discipline; and service to the university, the profession, and the community at large. The standards are drafted by the unit’s Committee on Promotion and Tenure, and academic unit heads supply copies of these standards with any amendments and revisions to the Provost for approval. A copy of the appropriate set of standards is given to each faculty member by the Office of the Provost at the time of the faculty member’s initial appointment.

(b) The process used to determine faculty salaries
   The available salary adjustment funds are allocated to the deans by the provost. In Armour College of Engineering the dean meets with the department heads that propose a distribution of these funds to their faculty according to the faculty activities and evaluations for the preceding year, and the needs of the departments/programs. The proposed adjustments are approved by the provost.

(c) Faculty Workload
   The official workload for a full-time faculty member is 9 contact hours or equivalent per semester over a 2 semester academic year. This requirement can be met through: teaching of regularly scheduled courses; advising either or both undergraduate and graduate students; serving as research mentor for undergraduate or graduate students with a project, thesis, or dissertation outcome with the general rule that ten student credit hours is equivalent to one contact hour; developing new or redesigning existing courses and/or laboratories; or teaching courses with a large enrollment or highly intensive laboratory or project component.

   Other activities that carry equivalent teaching credit include: serving as chair or associate chair of academic units, acting in other defined administrative roles within academic units, or providing extensive committee service for the academic unit, college, or university. In general, research funds can be used to reduce a faculty member's teaching
load with the general guideline that approximately one month of academic year salary is equivalent to two contact hours.

In addition, varying contact hour credit is given in certain special cases such as teaching a course for the first time, teaching a distance learning course, or supervising seminar courses.

Untenured faculty members on the tenure track are provided with a minimum credit per academic year of six contact hours reduction in their teaching loads and may receive up to twelve contact hours of reduced teaching at the discretion of their academic unit head.

(d) Faculty Benefits
Faculty benefits include: mandatory individual and family health insurance program (Blue Cross/Blue Shield); life and permanent disability insurance; university matched 503(b) retirement program; optional dental insurance; and a tuition remission program for family members of the faculty. Faculty are eligible to apply for sabbatical leave after each 6 years of service.

(e) Supervision of Part-time Faculty
Part-time faculty are hired by the academic units after an interview process and evaluated on a semester-by-semester basis by the unit head or a designee. Student teaching evaluations identical to those used for full-time faculty are conducted for each course taught by a part-time faculty member. These evaluations, along with other inputs, are used by the unit head to determine the teaching performance of part-time faculty. On occasion, the academic unit head or a designee will attend one or more classes taught by part-time instructors to evaluate their performance directly. The course outline and textbook selection is made by the cognizant full-time faculty member who normally teaches the specific course.
University Academic Structure
Credit Unit

One semester credit hour represents one class hour or three laboratory hours per week. One academic year represents 30 weeks of classes, exclusive of final examinations.

Instructional Modes

Non-traditional modes of instruction are not employed in the undergraduate engineering programs as a general rule.

Grade-Point Average and Graduation Requirements

A four point grading scale is used, with “A”=4, “B”=3, “C”=2, “D”=1 and “E” (fail) = 0. A grade point average of 2.0 cumulative and a 2.0 average in courses designated as major courses is required to graduate.

The Office of Educational Services is responsible for certifying that an individual student has satisfied the prescribed curriculum for a Bachelor of Science degree in engineering. When necessary, the associate chair of the individual engineering department provides assistance in the verification process.

An academic audit provides a summary of a student’s academic status to date and lists the courses to be completed in order to receive a degree. Engineering students who have completed at least 60 semester hours (including applicable transfer credit) will receive an audit from the Office of Educational Services. After receiving their first audit, students may request periodic updates. Faculty advisors have access to the same database of student information that is used by the Office of Educational Services.

After a student submits an application for graduation, a graduation audit is completed and a letter, which indicates the remaining requirements for the degree, is sent to the student. The final audit is completed when the grades for the semester are recorded and, if all requirements are completed, the degree is awarded.

Academic Supporting Units

The Department of Applied Mathematics teaches required courses in calculus and differential equations. The department head is Dr. F. Hickernell.

The Department of Biological, Chemical and Physical Sciences teaches required courses in Physics, Biology and Chemistry. The department head is Dr. J. Zasadzinski.

Non-Academic Supporting Units
The Academic Resource Center (ARC)

The ARC supports many required undergraduate courses in mathematics, physics, and chemistry, and many lower division engineering courses.

The ARC hires 22-30 tutors a semester, depending on how many hours each tutor works. For tutors to work in the ARC, they must be a rising junior or senior, with a 3.5 GPA in the major they wish to tutor. Tutors are recruited by faculty referral, current tutor recommendations, advertising in IIT Today, or by running GPA reports of students by major and year. After students turn in a faculty letter of recommendation and application, they interview with both the director and a subject-specific tutor, who asks them to answer questions in a mock-tutoring session that are representative of the questions we get in the ARC.

In addition to one-on-one and group tutoring, the ARC also runs examination reviews by student or professor request. This past school year, the ARC ran review sessions in PHYS 123 and 221, as well as MMAE305 and MMAE320. During the fall 2008 semester, the ARC had 4,000 visits.

There is one permanent staff member:
Dr. Elizabeth Lyons, Director, Academic Resource Center (2007)

Education: MFA in Creative Writing, Purdue University, 2006
BA in English, College of Charleston, 2003

Prior Employment History:
Assistant Director, Academic Resource Center (2006-2007)
Graduate Instructor, Purdue University, 2003-2006
Writing Lab Coordinator and Tutor, Purdue University, 2004-2006

IIT Writing Center

Individual assistance for writing assignments is available in the IIT Writing Center, which provides guidance with assignments in engineering, science, and technical communication courses as well as courses in the humanities (literature, history, art & architecture history, philosophy) and social sciences.

One-on-one instruction focuses on the specific needs of the individual student. Typically, a student takes a project or paper assignment to the Writing Center, where a tutor assists with the writing process:

- by helping to interpret the goals and requirements of the assignment
- by guiding the processes of information gathering, analyzing, evaluating, synthesizing, organizing, and documenting

Appendix I - 9
• by helping to address "local" issues such as grammar, punctuation, spelling, conventions of typing, etc.

The Writing Center is opened four days a week, Monday through Thursday, with a typical daily schedule of 9:30 AM to 3:30 pm on Mondays and Wednesdays, and 10:00 am to 5:00 pm on Tuesdays and Thursdays. Weekly student sign-up sheets were posted on the faculty offices of 232 and 233, allowing visitors to make their own appointments. Most appointment lasted from one half to one hour. Students may bring in any form of writing, from an undergraduate first-year composition assignment to a PhD project.

The total number of Spring 2008 visitors was 98. Non-native speakers totaled 67, native English speakers 31. Undergraduates totaled 69 and graduates 29. Total tutoring sessions numbered 236. More that one third (31 of 98 total) of the students were native English Speakers.

There is one permanent staff member:
James Dabbert, Director, IIT Writing Center, 1997-2008
Senior Lecturer, Lewis Department of Humanities, Illinois Institute of Technology, 2000-2007

Education: B.A., English, Indiana University, Bloomington, Indiana, 1967
M.S., Linguistics, Indiana University, Bloomington, Indiana, 1977

Prior Appointments:
Instructor, Lewis Department of Humanities, Illinois Institute of Technology, 1989-2000

Office of Technology Services (OTS)

Main Campus Infrastructure
(a) Academic Buildings

From 2006 to 2008, OTS has upgraded technology at Engineering 1 (E1), Stuart, Metals, Siegel Hall, Life Sciences, Perlstein, and Crown Hall buildings. These recent upgrades include:

Fiber connection into the buildings
Fiber raiser between floors
New teledata closets
New network switches
Infrastructure for Distance Learning
A/V equipped classrooms/labs
Full wireless coverage
Replacement of CAT3 cables with CAT6 cables

Appendix I - 10
UPS in teledata

Engineering and computer science classroom and lab work activities are usually conducted in the following buildings: Stuart, E1, Alumni Hall, and Siegel Hall, each of which benefited from these upgrades.

(b) Classrooms

IIT offers three levels of technology enhanced classrooms:
Basic A/V classroom, which is equipped with a network connection, a projector and screen, an ELMO and a VHS/DVD deck. All components are controlled through a single Crestron Control Panel on the instructor's desk.
Distance Learning Classroom has all the equipment of a basic A/V classroom, plus one or two video cameras, instructor and student microphones, plasma TV monitor, connections to broadcasting and digitizing devices for TV and/or Internet delivery. These classrooms also broadcast via television and the Internet.
Video Conferencing Classroom, which is similar to Distance Learning Classroom but also allows for real-time collaboration with a remote classroom location.

In addition, a PC Classroom is an OTS computer lab that is equipped with a PC and projector for the instructor and individual computers for each student. This arrangement provides students with a hands-on learning experience.

The following buildings are equipped with technology-enhanced learning classrooms:

Stuart Building:
8 basic A/V classrooms
8 distance learning classrooms (2 of which are videoconferencing classrooms)
4 PC classrooms

E1:
14 basic A/V classrooms
3 distance learning classrooms
1 PC classroom

Alumni Hall:
2 basic A/V classrooms
1 PC classroom

Siegel Hall:
1 basic A/V classrooms
2 PC classrooms

(c) OTS Computer Labs:
OTS operates 12 labs in the Main Campus. The Stuart Building, E1, Alumni Hall and Siegel Hall computer labs were the focus of technology upgrades in 2006-2007 or are scheduled for upgrades within the next year. The Engineering and computer science student community usually use the labs in the following buildings:

The E1 building computer lab in room 029 has 21 workstations.

E1 029:
Equipped with basic A/V System in Summer 2006.
The 21 PCs that were refreshed in 2007 are due to be refreshed in Summer 2010.

The Stuart Building has four computer labs, with a total of 109 workstations.

Stuart 112J:
Equipped with basic A/V System in Summer 2006.
The 46 PCs were refreshed in 2005 are due to be refreshed Summer 2008.

Stuart 112E:
The 22 PCs that were refreshed in 2006 are due to be refreshed in Summer 2009

Stuart 112F:
The 22 PCs that were refreshed in 2006 are due to be refreshed in Summer 2009.

Stuart 112X: (An open work area)
The 19 PCs that were refreshed in 2006 are due to be refreshed in Summer 2009, with the addition of 4 new workstations.

Alumni Hall has one computer lab with 29 workstations:

Alumni 218:
The 29 PCs that were refreshed in 2007 are due to be refreshed Summer 2010.

Siegel Hall has two computer labs with 52 workstations:

Siegel 237:
The 31 PCs that were refreshed in 2006 are due to be refreshed Summer 2009.

Siegel 236:
The 21 PCs that were refreshed in 2007 are due to be refreshed Summer 2010.

The MTTC Night Owl Lab, opened in February 2006:
The 50 laptops are due to be refreshed Summer 2008.

(d) Software

OTS PC labs offer 81 current software titles that specifically address engineering students' needs, and 49 titles that are geared toward Computer Science students. These titles are reviewed every semester by the IIT Software Committee, and are updated after thorough testing for compatibility with existing lab hardware/software.
(e) Distance Learning

IIT Online provides technology and procedural training for all new distance learning faculty. This is primarily utilized in Masters and certificate programs and will not be described here.

(F) Blackboard

Since 2003, the number of courses utilizing the Blackboard course management system has increased six fold. The Blackboard system hosts a website for every course offered at IIT and serves as a portal to IIT Online streaming media, which can be accessed by students in both online and live course sections. Instructors post notes, lectures and assignments on the course page, which also features a discussion board and chat room.

In Fall 2008, 161 of the 579 (27.8%) Engineering courses use the Blackboard management system.

Each Fall, OTS conducts group Blackboard training for new professors. New professors arriving in Spring and Summer are offered either group or individual Blackboard training. Advanced Blackboard training sessions are also available for faculty currently using the system.

OTS operates under the direction of:

Ophir Trigalo, Chief Information Officer, 2003 – 2008

Education: M.B.A. Information Systems, Tel Aviv University, 1990
   B.A. Economics and Statistics, Ben Gurion University, 1983

Prior appointments
   Vice President for Information Services, Depaul University, 1997-2003

University Libraries

IIT libraries provide access to an extensive collection of print and digital resources in support of the institution’s academic disciplines including architecture, design, engineering, computer science, business, and law. There are six libraries at IIT located on four campuses. The Paul V. Galvin Library serves as the main library for the Illinois Institute of Technology and provides primary support for all programs in the fields of
engineering and computer science. The Downtown Campus Library serves IIT’s Chicago-Kent College of Law and the Stuart Graduate School of Business. Branch and departmental libraries include the Graham Resource Center serving the College of Architecture, the Louis W. Biegler Library on IIT’s Rice Campus, the Center for the Study of Ethics in the Professions Library, and the National Center for Food Safety and Technology Library.

Collections
Collectively, the libraries’ collections consist of over 1.8 million volumes, including books, journals, videos, DVDs, maps, microform, and government documents. In addition, the libraries provide 24/7 access to a broad range of digital resources including over 100 online databases, more than 24,000 full-text scholarly journal titles, and over 7,000 full text e-book titles in computer science and technology related fields.

A founding member of the Consortium of Academic and Research Libraries in Illinois (CARLI), IIT libraries also provide access to more than 32 million library items from 75 additional academic libraries statewide. Along with extensive resource sharing, IIT’s membership in CARLI enables IIT libraries to develop partnerships with over 140 Illinois libraries and take advantage of innovations in teaching, research, technology, and services as well as opportunities to enhance its collections. In 2007, IIT libraries were awarded several cooperative collection development grants in conjunction with other Illinois university libraries that have been used to enrich the libraries collections particular in the fields of science and technology. These specialized areas include “Applied Mathematics in Support of Homeland Security”, “Computer Science Mathematics and Computer Algorithms”, “Internet Telephony and Computer Crimes Investigation”, and a collection partnership in Green Manufacturing Technology.

While the Galvin Library supports the university’s core curriculum and all subject disciplines, IIT branch and departmental libraries also provide specialized collections of resources that directly and indirectly support science, technology, and engineering. In addition to its rapidly growing architecture collection, the library also contains materials of interest to those in related engineering fields. The Ethics Center Library has a growing collection of materials on practical and professional ethics including items related to ethical issues and activities in areas across the disciplines including computer science, engineering, and the sciences, as well as items addressing cross disciplinary issues relating to the professions such as confidentiality, conflicts of interest, and professional concerns such as self-regulation and continuing education. The Ethics Center Library also maintains the most comprehensive online collection of codes of ethics in the world as well as a variety of print and online resources, including the “NanoEthicsBank”, an online, annotated bibliography of materials developed by IIT’s Center for the Study of Ethics in the Professions, that includes reports, regulatory documents, codes of ethics, research and development, and other resources related to nanotechnology and nanoparticles.
IIT libraries are actively engaged in the ongoing assessment of the quality and currency of its print and digital collections in order to meet the increasing demand of the growing student population as well as support emerging curriculum needs. In 2004 and 2006, the Galvin Library participated in LibQUAL, a library service and quality assessment process, in order to evaluate faculty, staff, and student satisfaction with IIT libraries’ collections, services, and facilities, as well as to monitor the impact changes made in response to the 2004 assessment had on current user satisfaction.

As a result of the assessment and additional collection analysis that identified potential areas requiring additional attention, the IIT libraries began an initiative to review and reconstitute its core monograph collections which includes five-year goals for the development of the print and online collections. In response to user assessment, the book acquisitions funding formula for the main library was also redesigned to increase expenditure in various disciplines such as the basic sciences and mathematics in order to more adequately support programs that are part of the undergraduate core curriculum.

Due to the renewed focus on the collections and the significant increase in funding allocated for monograph acquisitions, the Galvin Library’s print collection has substantially improved over the last five years particularly in the areas of science, technology, and engineering. The number of computer science titles purchased in FY2007 was almost twice the amount purchased in FY2003. The number of titles purchased overall in the fields of science, technology, and engineering in FY2007 was three times the amount purchased in FY2003.

IIT libraries also created new collection development policies to foster the development of more contemporary monograph collections and shift away from a focus heavily weighted towards the development of traditional programs to also include new disciplines as well. With additional support from the university and this new direction, Galvin Library was able to build foundation collections to support IIT’s new Biomedical Engineering program. To identify the unique collection needs across the disciplines, all libraries also employ an active departmental liaison program, staffed with subject specialists having skills unique to each program, and these liaisons consult with faculty on resources that will contribute to the development of library collections in support of their curriculum.

IIT libraries also have undertaken a comprehensive review of current print journal subscriptions, resulting in a transition at the main library from a primarily print-based journal collection of a few thousand titles to a primarily online journal collection of over 24,000 full-text titles, 8,400 of which are in the sciences and technology. These titles not only include individual subscriptions but also include multiple titles provided through publisher “bundles” including ACM, ACS, ASCE, ASME, and IEEE. Online database access has also shifted from providing numerous small, lower quality services to selecting the “best of class” in each discipline, resulting in subscriptions to INSPEC, CSA Technology Research Database, COMPENDEX, Web of Science, and SciFinder.
Scholar. In 2007, the SIAM Locus Journal Archive, Institute of Mathematics Statistics Journals, and Wiley Interscience Electronic Journals were also added to the collection of electronic resources available to the IIT community.

Services/Innovative Technology
IIT libraries are particularly well known for their use of innovative technology to support student learning and effective teaching. The libraries were among the first in the country to implement an electronic reserves system; web-based document delivery for interlibrary loan; remote access to a diverse collection of digital resources; wireless networking; and a laptop loaner program. The libraries provide ongoing support for digital resources and information technology through a long-term commitment of library personnel, technology, and technological expertise which contributes to the development and expansion of resources and information technology centered services unique to libraries and the communities they serve. In addition to financial support provided by the university, IIT libraries – particularly Galvin Library – have received several state and federal grants in support of library technology initiatives and continue to seek additional funding for the development of emerging technologies and technology-based services to better serve the changing needs and expectations of its users for less traditional methodologies of information access, retrieval, and dissemination.

IIT libraries also continue to offer innovative services that use new technologies to facilitate communication between students and librarians. Along with in-person consultations, Galvin Library offers access to reference librarians through email and instant messaging which has become a popular and efficient way of getting expert assistance quickly. In 2007, over 18% of Galvin Library’s total reference transactions occurred electronically through IM or email contact. The Galvin Library also introduced a library blog in 2007 to keep library users informed of new resources, collections, and services available to them as they become available.

Over the last several years, the libraries have also significantly increased public computing resources in response to user demand. The Galvin Library, in particular, has experienced a significant growth in on-site use of library resources over the past several years and has continued to add additional public workstations to meet this demand as illustrated by the estimated 20% increase in public workstation logins at the main library between 2006 and 2007. Annual visitors to the Galvin Library was over 205,000 in 2007 which represents a 48% increase over the 139,000 total visitors in 2003.

Instruction
Emerging technologies have also been employed by IIT libraries to promote innovative and interactive instruction in support of the curriculum. The Library Learning Center (LLC), a state-of-the-art learning resource center on the lower level of the Galvin Library that opened in 2000, continues to foster a highly adaptable and collaborative teaching and interactive learning environment by employing the latest information resources and technology. The LLC is used extensively for traditional bibliographic and library skills
Appendix I - 17

instruction on the main campus as well as increasingly more collaborative, problem-based information literacy instruction that focuses on developing skills that will more fully support students’ academic growth as well as their long-term professional development.

This collaborative instruction approach, in which librarians work with faculty to create contextual, course-specific assignments and instruction materials, was developed to fill a need for students in engineering programs to possess information literacy skills that are developed in the classroom, improved through their research, and then continued on into the profession.

Galvin Library’s role in this development within the academic environment is to provide resources that will aid and services that will instruct students in identifying, locating and effectively using information. The senior engineering librarian, as well as other subject specialists in engineering and the sciences, present instruction in library resources during Introduction to the Profession (ITP) classes that are mandated for all students in the engineering disciplines and for other classes upon request. ITP classes expose students to concepts that are part of the training necessary to acquire information literacy skills that will lead them through their early formative academic years, and create an acknowledgement of the need to engage in life-long research and learning in their profession. In addition, a regular variety of library workshops are offered that expand upon ITP classes and are more specific in the content presented, such as patents, standards and technical reports. These sessions are attended by both graduate and undergraduate students alike.

In Fall 2007, a new structure for collaborative or blended instruction was put into practice for CAEE classes. Librarians worked with faculty in designing three assignments weeks apart and with library instruction in between that were context specific to course materials and therefore more meaningful to the student. Assessment showed that the students agreed the instruction was effective in improving the quality of their research and assignments. Additional collaborative opportunities in other courses will be pursued by librarians and further assessment of library instruction will be conducted.

IIT Libraries operate under the direction of:

Christopher Stewart, Dean of Libraries, 2006 - 2008

Education: BA, Political Science, University of Illinois at Chicago, 1989
MLS, Dominican University, 1995
MBA, Illinois Institute of Technology, 2004
Ed.D, University of Pennsylvania (expected, 2009)

Prior Appointments
2004-2006, Acting Dean of Libraries, Illinois Institute of Technology

Appendix I - 17
2002-2004, Associate Dean of Libraries, Illinois Institute of Technology
2000-2002, Associate Dean for Library Technology, Illinois Institute of Technology
1999-2000, Associate Dean for Network Services, Paul V. Galvin Library, IIT
1998-1999, Assistant Dean for Network Services, Paul V. Galvin Library, IIT
1995-1998, Network Services Manager, Paul V. Galvin Library, IIT

Career Management Center

The Career Management Center (CMC) at Illinois Institute of Technology serves the critical function of providing the linkage between students and graduates with local, national and international employers. The CMC’s mission is to engage students and alumni to develop and practice lifelong career management skills to realize their career goals. Students are strongly encouraged to register with the CMC during their freshman year in order to begin developing their careers as soon as possible. The CMC also seeks to develop lasting partnerships with employers by providing employers with the opportunity to participate in key programs to identify, and hire skilled and technically prepared individuals. The Director of CMC is Bruce Mueller, and CMC has a professional staff of eight.

CMC Programs

Career Fairs: IIT Career Fairs are open to all local, national and international employers seeking quality hires from all disciplines. The Illinois Institute of Technology is a prime institution targeted by many employers. The Career Management Center sponsors two Career Fairs each school year.

On-Campus Interviewing: On-Campus Interviewing (OCI) is a program allowing employers can use to interview and hire IIT graduates, alumni, and undergraduates seeking full-time, co-op or internship positions on IIT’s campuses. The Career Management Center holds OCI during the fall and spring semesters. The fall session runs from mid-September through early December and the Spring session runs from mid-February through early May.

Job Listings: Employers may post job listings on the CMC’s eRecruiting website.

Resume Development: CMC provides workshops and one-on-one advising on resume writing throughout the year. Students and alumni may post their resumes on eRecruiting, where they are made available to potential employers.

Cooperative Education & Internship Programs: CMC provides monitoring and administrative services for students in approved Cooperative Education & Internship positions. Assistance in obtaining Curricular Practical Training (CPT) Work
Authorization for these positions is also provided to international students by the CMC in partnership with International Center.

Tracking Reports: CMC tracks graduating students’ progress in finding employment or enrollment in post-graduate programs, and provides a regular summary report to the academic units.

Web Site: CMC maintains a comprehensive web site with online resources for students, alumni and employers at www.cmc.iit.edu

Professional Development Programs & Workshops: CMC provides several programs and workshops to help students develop professionally. The programs include one-on-one career advising with a Career Counselor who specializes in the student’s field, mock interviews, and resume and cover letter critiques. Workshops include the Getting a Job three-part series, Making a Positive First Impression, Marketing Yourself Effectively and Transitioning from Student to Professional, Etiquette Lunches or Dinners, and Dress for Success.

The CMC operates under the direction of:


Education: BBA (1968), MBA (1974)

Prior Appointments:
ACS-Managing Director, Global Human Resources Outsourcing (2002-2005)
Motorola-Corporate Vice President, Human Resources Infrastructure and Technology (1983-2003)

The Interprofessional Projects (IPRO) Program Office

The IPRO Program Office is responsible for administering and coordinating all aspects of The IIT Interprofessional Projects (IPRO) Program. This office was established in 1995 in order to plan for and eventually implement the general education requirement that all undergraduates complete two interprofessional project courses in order to graduate, with each course representing three credit hours. There are two prominent functions that achieve this result:

1. Since the interprofessional course, by design, encompasses all professional disciplines and programs, the IPRO Program Office has the responsibility to coordinate and integrate faculty, sponsors and students in order to identify, organize, promote, implement and assess approximately 90 IPRO project course sections (i.e., project teams) each year so that our students can fulfill their interprofessional project requirement. This serves on the
order of 1,000 students each year (producing on the order of 3,000 credit-hours), with an
average team size of eleven students from any level (sophomore through graduate) and
any discipline and professional program at IIT, although the vast majority of students
have junior or senior standing. The disciplines involved across all IPRO course sections
encompass all undergraduate degree programs: applied mathematics, architecture,
business, computer science, engineering (aerospace, architectural, biological, biomedical,
chemical, civil, computer, electrical, environmental, materials, mechanical), the sciences
(biology, chemistry, physics), humanities (internet communication, journalism, technical
communication), industrial technology and management, information technology and
management, math and science education, psychology and social sciences (political
science, public administration). Graduate students may also participate and receive credit
toward their degrees, depending on their field of study, including, in addition to those
previously mentioned, law, design, and food safety and technology.

2. Since the purpose of the interprofessional course is to provide students with experiences
that emulate the workplace, an important aspect of the IPRO Program is the involvement
of workplace organizations that identify viable “real world” complex topics, and provide
financial support and professional advice to our IPRO teams throughout the semester.
One-third of projects are currently sponsored, with a long-term goal of achieving two-
thirds sponsorship, although many projects already benefit from informal collaboration
with a range of business, non-profit, entrepreneurial and public sector organizations.

The roles and responsibilities of The IPRO Program Office are thus summarized as
follows:

- Facilitate review and implementation of policies and procedures that define the learning
  objectives and govern the fulfillment of the two-IPRO project course general education
  requirement.
- Implement and maintain an efficient and effective system for creating, delivering and
  assessing project courses consistent with the learning objectives established for an
  interprofessional project experience.
- Develop and maintain sponsor relationships that are compatible with our faculty expertise
  and offer interesting and challenging learning experiences for our students, and that provide
  financial resources to help support the costs of coordinating and delivering the
  interprofessional project experience in a professional manner.
- Manage an operating budget and various grant and unrestricted donation accounts that
  support the delivery of the interprofessional course.
- Organize various events that support the learning objectives, including workshops (e.g.,
teambuilding, communication, project management, ethical decision-making, business
planning).
- Organize and participate in various events that support the development and advancement of
  the interprofessional curriculum, including faculty orientation sessions, faculty development
  workshops and other education conference opportunities that help to promote information
  exchange between IIT faculty and colleagues at other institutions, particularly in the field of
  team project based learning modalities.

Appendix I - 20
• Participate in open houses for prospective students and organize presentations and conferences related to interprofessional education (e.g., Best Practices of Interdisciplinary Team Project Programs, presentations to sponsors and trustees)
• Coordinate the end-of-semester IPRO Projects Day Conference (held three times each year) that provides a venue for all IPRO teams to present their work via formal oral presentations and interactive exhibits, and includes a judging process (with working professionals, faculty members and graduate students) that is linked to assessment of learning objectives, and offers a showcase event for IIT alumni, trustees, sponsors, employers, high school and junior high school students, parents of IIT students and prospective students, and the general public.
• Support the information needs of and be responsive to the Interprofessional Studies Committee, a committee established via the University Faculty Council to provide academic oversight of the IPRO course.
• Coordinate the IPRO proposal review process each semester that leads to the review of candidate IPRO projects for the subsequent semester, with on the order of half of the IPRO projects continuing and half new each semester.
• Identify and encourage the use of best practices by IPRO instructors that have value in enhancing the effectiveness of IPRO teams (e.g., peer evaluation, grading guidelines (team performance and individual performance on the team).
• Encourage academic units and faculty to collaborate across disciplines and programs, recognize innovative approaches and support scholarship, publication and presentation at national conferences.
• Encourage graduate students to participate on interprofessional project teams and seek ways to adapt the IPRO course model to support graduate research and commercialization activities across professional boundaries and build competency of graduate students to team teach and teach in teams.
• Integrate and coordinate the process for students to enroll in interprofessional courses as part of the regular course registration schedule and provide timely information about IPRO course topics at http://ipro.iit.edu, giving particular attention to constraints that help to control the size of the team and the mix of students from various disciplines on a team.
• Provide a syllabus template that offers a generic framework and semester schedule for IPRO instructors to use in planning and implementing an IPRO project course.
• Coordinate IPRO team tools that facilitate communication and record keeping, including http://igroups.iit.edu and http://iknow.iit.edu.
• Coordinate the submittal and review of deliverables by IPRO teams, including: project plan, code of ethics, mid-term review, web site (optional), final oral presentation, exhibit/poster, abstract, final report and team work product.
• Coordinate surveys and evaluation tools that provide feedback to the IPRO Program Office from students, faculty, sponsors and alumni.
• Provide IPRO stipends to support the assignment of IIT faculty members in serving as an IPRO instructor as part of their regular teaching load, support part-time IPRO instructors who offer specialized expertise and capacity and support IPRO team expenses on an as-needed basis.
• Coordinate with IIT’s director of entrepreneurship and the Jules F. Knapp Entrepreneurship Center to encourage student and faculty ideas for Entrepreneurial IPRO (EnPRO) projects that meet all of the requirements of a typical interprofessional project and encompass venture development and opportunity analysis that can lead to a business plan, prototype and user testing.

• Coordinate with the IIT Leadership Academy in delivering various teambuilding and leadership seminars and workshops.

• The IPRO Program Office is supported as follows: Director of Interprofessional Studies and The IPRO Program (full time), Associate Provost for Undergraduate Affairs (20%), IPRO Administrative Assistant/Coordinator (full time) and part-time graduate and undergraduate students.

Thomas M. Jacobius, Director, Interprofessional Studies & The IPRO Program, 2000-2008

Education:  BS, Mechanical Engineering, IIT, 1971  
MBA, Northwestern University, 1978

Prior appointments:
1995-2000  Director, Industrial Liaison & Technology Transfer; Co-Director, The IIT Interprofessional Projects (IPRO) Program, IIT.
1991-1995  Director, Office of Research Admin and Office of IP Management, IIT.
1988-1989  Program Manager, Rail Simulation & Training Group, IIT Research Institute.
Table 1-1. History of Admissions Standards for Freshmen Admissions for Past Five Years

<table>
<thead>
<tr>
<th>Computer Science</th>
<th>Fall of Academic Year</th>
<th>Composite ACT</th>
<th>Composite SAT</th>
<th>Number of New Students Enrolled</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>2007-8</td>
<td>MIN. 20 AVG. 29</td>
<td>1000 AVG. 1292</td>
<td>39</td>
</tr>
<tr>
<td></td>
<td>2006-7</td>
<td>21 29</td>
<td>970 1320</td>
<td>37</td>
</tr>
<tr>
<td></td>
<td>2005-6</td>
<td>22 28</td>
<td>1030 1283</td>
<td>43</td>
</tr>
<tr>
<td></td>
<td>2004-5</td>
<td>21 28</td>
<td>860 1288</td>
<td>42</td>
</tr>
<tr>
<td></td>
<td>2003-4</td>
<td>19 28</td>
<td>860 1290</td>
<td>31</td>
</tr>
</tbody>
</table>

Table 1-2. Transfer Students for Past Five Academic Years

<table>
<thead>
<tr>
<th>Computer Science</th>
<th>Fall of Academic Year</th>
<th>Number of Transfer Students Enrolled</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>2007-8</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>2006-7</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>2005-6</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>2004-5</td>
<td>9</td>
</tr>
<tr>
<td></td>
<td>2003-4</td>
<td>10</td>
</tr>
</tbody>
</table>

Table 1-3. Undergraduate Enrollment Trends for Past Five Academic Years

<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Enrollment during Fall</td>
<td>195 172 160 159 157</td>
</tr>
<tr>
<td>Full-time Students</td>
<td>16 8 10 13 10</td>
</tr>
<tr>
<td>Part-time Students</td>
<td>209.7 180.5 169.5 166.2 163.5</td>
</tr>
<tr>
<td>Completions between 7/1 and 6/30</td>
<td>69 39 48 29 20</td>
</tr>
</tbody>
</table>

1 FTE = Full-Time Equivalent: 15 Credit hours = 1FTE
2007-8 Graduate value includes ONLY Summer and Fall, not Spring as those values are not yet available.
Table 1-4. Program Graduates

*Computer Science*
(For Past Five Years or last 25 graduates, whichever is smaller)

<table>
<thead>
<tr>
<th>Numerical Identifier</th>
<th>Year Matriculated</th>
<th>Year Graduated</th>
</tr>
</thead>
<tbody>
<tr>
<td>10229632</td>
<td>2002 Fall</td>
<td>2007 Fall</td>
</tr>
<tr>
<td>10229842</td>
<td>2003 Fall</td>
<td>2007 Fall</td>
</tr>
<tr>
<td>10239544</td>
<td>2002 Fall</td>
<td>2007 Fall</td>
</tr>
<tr>
<td>10241171</td>
<td>2003 Fall</td>
<td>2007 Fall</td>
</tr>
<tr>
<td>10295257</td>
<td>1980 Fall</td>
<td>2007 Fall</td>
</tr>
<tr>
<td>10372138</td>
<td>2003 Fall</td>
<td>2007 Fall</td>
</tr>
<tr>
<td>10387963</td>
<td>2004 Fall</td>
<td>2007 Fall</td>
</tr>
<tr>
<td>10393125</td>
<td>2005 Spring</td>
<td>2007 Fall</td>
</tr>
<tr>
<td>10393682</td>
<td>2004 Fall</td>
<td>2007 Fall</td>
</tr>
<tr>
<td>10396582</td>
<td>2004 Fall</td>
<td>2007 Fall</td>
</tr>
<tr>
<td>10397138</td>
<td>2005 Fall</td>
<td>2007 Fall</td>
</tr>
<tr>
<td>10401711</td>
<td>2005 Spring</td>
<td>2007 Fall</td>
</tr>
<tr>
<td>10415148</td>
<td>2006 Fall</td>
<td>2007 Fall</td>
</tr>
<tr>
<td>10206760</td>
<td>2003 Fall</td>
<td>2007 Summer</td>
</tr>
<tr>
<td>10238006</td>
<td>2003 Fall</td>
<td>2007 Summer</td>
</tr>
<tr>
<td>10248253</td>
<td>2005 Fall</td>
<td>2007 Summer</td>
</tr>
<tr>
<td>10291282</td>
<td>2003 Fall</td>
<td>2007 Summer</td>
</tr>
<tr>
<td>10370649</td>
<td>2002 Fall</td>
<td>2007 Summer</td>
</tr>
<tr>
<td>10394456</td>
<td>2004 Fall</td>
<td>2007 Summer</td>
</tr>
<tr>
<td>10394667</td>
<td>2004 Fall</td>
<td>2007 Summer</td>
</tr>
<tr>
<td>10209325</td>
<td>2003 Fall</td>
<td>2007 Spring</td>
</tr>
<tr>
<td>10210674</td>
<td>2003 Fall</td>
<td>2007 Spring</td>
</tr>
<tr>
<td>10212229</td>
<td>2003 Fall</td>
<td>2007 Spring</td>
</tr>
<tr>
<td>10229311</td>
<td>1993 Spring</td>
<td>2007 Spring</td>
</tr>
<tr>
<td>10238074</td>
<td>1999 Fall</td>
<td>2007 Spring</td>
</tr>
</tbody>
</table>

(NOTE: ABET recognizes that current information may not be available for all students)
### Table 1-5. Personnel and Students

**Computer Science**

**Fall 2007**

<table>
<thead>
<tr>
<th>Position</th>
<th>FT</th>
<th>PT</th>
<th>FTE</th>
<th>Ratio to Faculty</th>
</tr>
</thead>
<tbody>
<tr>
<td>Faculty (tenure-track)</td>
<td>13</td>
<td></td>
<td>13</td>
<td>0.44</td>
</tr>
<tr>
<td>Graduate Assistants</td>
<td>56</td>
<td>19</td>
<td></td>
<td>0.64</td>
</tr>
<tr>
<td>Office/Clerical Employees</td>
<td>2</td>
<td></td>
<td>2</td>
<td>0.07</td>
</tr>
<tr>
<td>Other Faculty (excluding Student Assistants)</td>
<td>12</td>
<td>4</td>
<td>13</td>
<td>0.45</td>
</tr>
<tr>
<td>Professional</td>
<td>1</td>
<td></td>
<td>1</td>
<td>0.03</td>
</tr>
<tr>
<td>Research Assistants</td>
<td>1</td>
<td></td>
<td>1</td>
<td>0.03</td>
</tr>
<tr>
<td>Administrative</td>
<td>3</td>
<td></td>
<td>3</td>
<td>0.10</td>
</tr>
<tr>
<td>Undergraduate Student Enrollment*</td>
<td>180</td>
<td>15</td>
<td>188</td>
<td>6.41</td>
</tr>
<tr>
<td>Graduate Student Enrollment</td>
<td>299</td>
<td>201</td>
<td>422</td>
<td>14.38</td>
</tr>
</tbody>
</table>

*Includes all classes (freshmen, sophomore, junior, senior, etc)*

FTE calculation:
- Undergraduate=Hours/15
- Graduate=Hours/9
**Table 1-6. Program Enrollment and Degree Data**

*Computer Science*

<table>
<thead>
<tr>
<th>Enrollment counts in Fall of AY</th>
<th>Undergraduate Enrollment by Class</th>
<th>Total Undergrad</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1st</td>
<td>2nd</td>
</tr>
<tr>
<td>CURRENT 2007-8</td>
<td>FT</td>
<td>46</td>
</tr>
<tr>
<td></td>
<td>PT</td>
<td>4</td>
</tr>
<tr>
<td>1 2006-7</td>
<td>FT</td>
<td>39</td>
</tr>
<tr>
<td></td>
<td>PT</td>
<td>1</td>
</tr>
<tr>
<td>2 2005-6</td>
<td>FT</td>
<td>47</td>
</tr>
<tr>
<td></td>
<td>PT</td>
<td>1</td>
</tr>
<tr>
<td>3 2004-5</td>
<td>FT</td>
<td>48</td>
</tr>
<tr>
<td></td>
<td>PT</td>
<td>1</td>
</tr>
<tr>
<td>4 2003-4</td>
<td>FT</td>
<td>36</td>
</tr>
<tr>
<td></td>
<td>PT</td>
<td>2</td>
</tr>
<tr>
<td>5 2002-3</td>
<td>FT</td>
<td>44</td>
</tr>
<tr>
<td></td>
<td>PT</td>
<td>2</td>
</tr>
</tbody>
</table>

Give official fall term enrollment figures (head count) for the current and preceding five academic years and undergraduate and graduate degrees conferred during each of those years. The "current" year means the academic year preceding the fall visit.

FT--full time  PT--part time  Other Class includes…

Appendix I - 26
<table>
<thead>
<tr>
<th>Department</th>
<th>Rank</th>
<th>Number</th>
<th>High</th>
<th>Mean</th>
<th>Low</th>
</tr>
</thead>
<tbody>
<tr>
<td>Biomedical Engineering</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Assistant Professor</td>
<td>5</td>
<td>$77,443</td>
<td>$75,279</td>
<td>$71,050</td>
<td></td>
</tr>
<tr>
<td>Associate Professor</td>
<td>5</td>
<td>$88,518</td>
<td>$77,349</td>
<td>$48,347</td>
<td></td>
</tr>
<tr>
<td>Lecturer</td>
<td>1</td>
<td>$54,671</td>
<td>$54,671</td>
<td>$54,671</td>
<td></td>
</tr>
<tr>
<td>Professor</td>
<td>2</td>
<td>$157,671</td>
<td>$114,127</td>
<td>$70,583</td>
<td></td>
</tr>
<tr>
<td>Chemical and Biological Engineering</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Assistant Professor</td>
<td>2</td>
<td>$77,761</td>
<td>$75,297</td>
<td>$72,833</td>
<td></td>
</tr>
<tr>
<td>Associate Professor</td>
<td>5</td>
<td>$98,700</td>
<td>$86,176</td>
<td>$77,236</td>
<td></td>
</tr>
<tr>
<td>Lecturer</td>
<td>1</td>
<td>$63,355</td>
<td>$63,355</td>
<td>$63,355</td>
<td></td>
</tr>
<tr>
<td>Professor</td>
<td>13</td>
<td>$153,450</td>
<td>$102,594</td>
<td>$73,196</td>
<td></td>
</tr>
<tr>
<td>Civil, Architectural and Environmental Engineering</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Assistant Professor</td>
<td>3</td>
<td>$76,482</td>
<td>$72,675</td>
<td>$68,549</td>
<td></td>
</tr>
<tr>
<td>Associate Professor</td>
<td>3</td>
<td>$76,381</td>
<td>$71,642</td>
<td>$67,384</td>
<td></td>
</tr>
<tr>
<td>Lecturer</td>
<td>3</td>
<td>$67,821</td>
<td>$61,251</td>
<td>$55,931</td>
<td></td>
</tr>
<tr>
<td>Professor</td>
<td>3</td>
<td>$112,063</td>
<td>$100,214</td>
<td>$84,137</td>
<td></td>
</tr>
<tr>
<td>Computer Science</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Assistant Professor</td>
<td>4</td>
<td>$85,067</td>
<td>$80,822</td>
<td>$69,785</td>
<td></td>
</tr>
<tr>
<td>Associate Professor</td>
<td>9</td>
<td>$102,000</td>
<td>$90,380</td>
<td>$65,644</td>
<td></td>
</tr>
<tr>
<td>Instructor</td>
<td>5</td>
<td>$62,732</td>
<td>$50,174</td>
<td>$27,159</td>
<td>(part-time)</td>
</tr>
<tr>
<td>Lecturer</td>
<td>4</td>
<td>$66,454</td>
<td>$64,822</td>
<td>$62,352</td>
<td></td>
</tr>
<tr>
<td>Professor</td>
<td>6</td>
<td>$157,120</td>
<td>$104,745</td>
<td>$21,976</td>
<td>(part-time)</td>
</tr>
<tr>
<td>Electrical Engineering</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Assistant Professor</td>
<td>10</td>
<td>$88,000</td>
<td>$83,951</td>
<td>$80,800</td>
<td></td>
</tr>
<tr>
<td>Associate Professor</td>
<td>4</td>
<td>$102,692</td>
<td>$90,217</td>
<td>$84,076</td>
<td></td>
</tr>
<tr>
<td>Lecturer</td>
<td>1</td>
<td>$53,000</td>
<td>$53,000</td>
<td>$53,000</td>
<td></td>
</tr>
<tr>
<td>Professor</td>
<td>7</td>
<td>$143,514</td>
<td>$112,980</td>
<td>$88,344</td>
<td></td>
</tr>
<tr>
<td>Mechanical, Materials and Aerospace Engineering</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Assistant Professor</td>
<td>4</td>
<td>$83,894</td>
<td>$79,172</td>
<td>$75,000</td>
<td></td>
</tr>
<tr>
<td>Associate Professor</td>
<td>9</td>
<td>$97,795</td>
<td>$87,743</td>
<td>$77,041</td>
<td></td>
</tr>
<tr>
<td>Lecturer</td>
<td>1</td>
<td>$36,828</td>
<td>$36,828</td>
<td>$36,828</td>
<td></td>
</tr>
<tr>
<td>Professor</td>
<td>8</td>
<td>$154,195</td>
<td>$111,867</td>
<td>$89,661</td>
<td></td>
</tr>
</tbody>
</table>

Salary figures do not include salary portions assigned to administrative duties.
APPENDIX II – Computer Science Department Policies and Procedures

CS Teaching Guidelines


The Computer Science Department understands and appreciates the time and commitment it takes to be an effective teacher. Our goal is to provide supportive infrastructure and uniform policies to insure that all courses are taught well and with common standards; this document outlines the responsibilities and standards for teaching courses in the department. It applies to all courses, undergraduate or graduate, required or elective, single section or multi-section, day or evening, Main Campus or Rice Campus, live or Internet or TV. It applies to all faculty, tenured or untenured, tenure-track or non-tenure-track, full time or part time.

This policy document is not a set of vague suggestions to instructors; it is a set of requirements for teaching in the department. The departmental tenure-track faculty have approved these requirements for all instructors and are committed to follow this policy themselves.

This document covers the following key areas: syllabus, course management, office hours, assignment and management of TAs, treatment of students, resources for students, cheating, student evaluations, archiving final exams, and submission of grades.

1 Syllabus
Instructors must provide a detailed syllabus for students delineating the objectives of the course. The objectives must substantially match those of the official course outline provided by the relevant course manager for a given course (for a list of course managers and course overviews see the web page www.cs.iit.edu/~abet): instructors have no flexibility in 100- and 200-level courses, very little flexibility in 300-level courses, limited flexibility in 400-level courses, and some flexibility in 500-level courses.

The syllabus must also indicate topics covered in the class, homework assignments, projects, exams, grading standards, and a clear policy on handling of late assignments/projects and academic irregularities (cheating—see below). The syllabus must be posted on the course home page linked from the CS web site no later than the end of the first week of the semester and, except for minor changes, must not be altered once published.

2 Course Management
Almost all courses have an assigned a course manager. The course manager is responsible for coordinating the activities of each section of the course. Instructors are obligated to meet and work with the course manager to insure the content and uniformity of courses both within a given semester and over the course of different semesters. Make sure to obtain the most current set of objectives for the course. The course manager will also provide guidelines for textbook choice, homework assignments, exams, grading, and student workload for the course.

In any semester, all instructors teaching a course must coordinate with one another and the course manager as to the course outline, text(s), exams, homework, and projects. Specifically, all sections of a course must use the same texts, reading lists, course outline, schedule of lectures, homework assignments, projects, and grading standards; all sections must use comparable
exams. Those teaching like-numbered courses must meet regularly to insure that the courses are progressing in parallel; they must work closely with one another in the assignment of final grades. In case of disagreements among those involved in a course, decisions will be made by the course manager or by the senior faculty member involved.

Plan to meet three times each semester with the Course Manager. The first meeting should be before the semester starts, to review the course syllabus, course policies, and course assignments to insure consistency. Meet again about halfway through the semester to check how things are going. Meet after the semester ends to discuss how things went, to review the “Course Self-Assessment by Instructor” and “Course Assessment by Students,” and to develop course enhancements.

While the choice of homework, projects, and exam questions is up to the instructor (subject to consistency across sections as outlined above), it is common sense to avoid reuse of problems, projects, or exam questions from prior semesters. Refresh your course each semester with new problems, projects, and exam questions.

3 Teaching Materials and Textbooks
No materials are to be reproduced and sold (even at cost) to students. Materials may be reproduced by the instructor and given to the students at no charge, or the materials may be placed on the course web page so that students can download and print them.

If a text authored by someone teaching for the department is required in a CS course, the royalties received by those authors for such texts sold at IIT should be donated to the department.

4 Office Hours
Establish and post reasonable office hours. Your office hours and location must be given on course web site; your office hours must be posted prominently on your office door. The location and times of office hours should match the location (main Campus or Rice) and times (day or evening) of the course. Be present in your office for all posted office hours. If you are teaching a course that includes part-time students, try to accommodate them by having some office hours on nights/weekends.

5 Grading Standards
Be certain that the students know what is expected of them. Establish grading standards for how much various components of the course (homework, quizzes, midterm exam, final exam, projects, and so on) will count in final grades, publicize those standards in class and on the course web site, and adhere closely to those standards. Do not change the standards once they are set and be careful to apply the standards equally to all students—to do otherwise is capricious grading.

6 Assignment and Management of TAs
Instructors of CS courses will be asked for recommendations for current TAs and prospective TAs at the end of each semester (around finals week). Preference in hiring TAs is given to Ph.D. students and prospective Ph.D. students, students currently doing research with tenure track faculty, and students with the highest grade point averages. Faculty and TA preference also plays a limited role; such preferences are taken into account in making assignments once the TAs are selected. The selection and assignment of TAs is a complicated job in which the overall well-being of the department must be considered, not just the desires of individual instructors or TAs.
You are responsible to be certain that TAs and graders handle their responsibilities. Monitor their performance and send an evaluation of it at the end of semester so that substandard TAs and graders can be eliminated.

TAs can do some teaching, hold office hours for student questions, assist in designing homework, projects, quizzes, and exams. Both TAs and graders can grade quizzes, homework assignments and student projects. In large courses, under appropriate supervision, TAs and graders can grade exams, but only if you treat their efforts as a first cut and personally review their work. You must also give graders a clear answer key and detailed guidance on the assignment of partial credit.

TA contact information and office hours should be posted on the course web page. TA office hours and location should match that of the course. Be sure to submit an evaluation of the TA(s) at the end of the semester.

7 Treatment of Students
Be reasonable in dealing with students. Be patient, compassionate, and courteous and be sure your TAs/graders are also. Treat students with a professional level of respect in all written and verbal communications. Give students frequent feedback on their performance on homework, projects, and exams throughout the semester.

8 Resources for Students
Instructors should be aware of the following IIT services for students and refer students to these services when appropriate:
Academic Resource Center—Tutoring for undergraduate courses, mostly 100-, 200-, and 300-level.
Student Health Center
Counseling Center—professional services to help students with academic, career, and personal concerns.
Academic Advisors—if any student is struggling or not attending class, especially for undergraduate students, please notify the student's Academic Advisor; see http://www.cs.iit.edu/faculty/undergrad_advisors.html

9 Cheating
Cheating is an occasional and a sad fact of life in the academic world. To insure consistency of policy across the department, if you encounter duplicate work (homeworks, projects, exams, and so on), take the following steps:
1. Make three copies of each instance of the duplicate work, one for your files, one to hand back to the students, and one for the departmental offices. Keep the original copies.
2. Deposit the copy for the departmental offices with the departmental assistant and include the students' names and ID numbers.
3. Assign each copy a grade of zero until you have talked to the students.
4. Talk to the students. If one student confesses to having stolen the work, or if you have clear evidence who is guilty, the guilty student should get a zero and the student who did the work should get full credit. If no one confesses, all involved should get a zero.
5. Inform students of the “second offense policy”: If a student is guilty of a second offense, the student will receive a grade of an E for the course and the department will pursue steps to remove him/her from the university.
A software comparison program called MOSS (www.cs.berkeley.edu/~moss) has been successfully used in CS105, CS106, CS200, CS331, CS401, CS402, and CS430. It is free to use and been used to narrow the number of student programs down that need to be hand reviewed for possible copying.

10 Student Evaluations
You are required to participate in the IIT Instructor and Course Evaluations and the department's self assessment studies at the end of every semester. Evaluations are done on line and you must encourage student participation. Self-assessment forms will be made available for distribution in classes near the end of the semester. Both the IIT course evaluations and the self-assessment forms are customarily reviewed by the Course Manager and the departmental administration; the course evaluations are also reviewed by the university administration.

11 Submission of Grades
Midterm grades must be submitted for all first year undergraduates in 100-, 200-, and 300-level courses. Midterm grades are used by Academic Advisors to identify students for early intervention.

Final grades must be computed following your published standards and submitted in a timely fashion, as required by the university. You must use Web for Faculty to submit grades online. You must also submit, in hard copy, your entire grade spreadsheet (showing grades for all assignments, exams, projects, and so on) to the departmental office: Grades for departmental files should be in two forms: (1) alphabetical by last name and (2) rank-ordered.

Grades of “I” (incomplete work) can be given only under limited circumstances: the student's work to date must be of passing quality and the student must have substantial equity in the course with no more than four weeks of coursework to be completed. See the IIT Bulletin for details:
http://www.grad.iit.edu/graduatecollege/programs

12 Archiving of Final Exams
All students’ final exam papers must be retained for one calendar year from the date of the exam. Part-time instructors and teaching assistants should deposit their exam papers, carefully labeled as to course, semester, and instructor, for storage in the department offices. Electronic archival is acceptable when appropriate.
**Mathematics**

<table>
<thead>
<tr>
<th>Course</th>
<th>Hrs.</th>
<th>Semester</th>
</tr>
</thead>
<tbody>
<tr>
<td>MATH 151</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>MATH 152</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>MATH 251</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>MATH 332 or 333</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>MATH 474 or 475</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>MATH Elective</td>
<td>3</td>
<td></td>
</tr>
</tbody>
</table>

**Total hours (23 hours minimum)**

Math Elective chosen from MATH252, MATH410, MATH453, MATH454, MATH476, MATH482

**Computer Science**

<table>
<thead>
<tr>
<th>Course</th>
<th>Hrs.</th>
<th>Semester</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS 100</td>
<td>(C) 2</td>
<td></td>
</tr>
<tr>
<td>CS 115*</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>CS 116*</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>CS 330**</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>CS 331</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>CS 350</td>
<td>(C) 3</td>
<td></td>
</tr>
<tr>
<td>CS 351</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>CS 430</td>
<td>(C) 3</td>
<td></td>
</tr>
<tr>
<td>CS 440</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>CS 450</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>CS 485</td>
<td>(C) 3</td>
<td></td>
</tr>
<tr>
<td>CS 487</td>
<td>(C) 3</td>
<td></td>
</tr>
<tr>
<td>CS</td>
<td>(C) 3</td>
<td></td>
</tr>
<tr>
<td>CS</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>CS</td>
<td>3</td>
<td></td>
</tr>
</tbody>
</table>

**Total hours (48 hours minimum)**

* CS 201 is a one-semester, accelerated course equivalent to the two-semester CS 115/CS116 course sequence.

**Science**

<table>
<thead>
<tr>
<th>Course</th>
<th>Hrs.</th>
<th>Semester</th>
</tr>
</thead>
<tbody>
<tr>
<td>PHYS 123</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>PHYS 221</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>Science Elective</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>Science Elective</td>
<td>3</td>
<td></td>
</tr>
</tbody>
</table>

**Total hours (14 hours minimum)**

**Humanities and Social or Behavioral Sciences**

<table>
<thead>
<tr>
<th>Course</th>
<th>Hrs.</th>
<th>Semester</th>
</tr>
</thead>
<tbody>
<tr>
<td>HUM 100-level</td>
<td>(C) 3</td>
<td></td>
</tr>
<tr>
<td>3xx (H)(C)</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>3xx (H)(C)</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>(S)(C)</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>3xx (S)(C)</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>(H)or(S)(C)</td>
<td>3</td>
<td></td>
</tr>
</tbody>
</table>

**Total hours (21 hours minimum)**

See the General Education Requirements

**Writing/Speaking**

<table>
<thead>
<tr>
<th>Course</th>
<th>Hrs.</th>
<th>Semester</th>
</tr>
</thead>
<tbody>
<tr>
<td>COM421 or 428</td>
<td>(C) 3</td>
<td></td>
</tr>
</tbody>
</table>

**Total hours (3 hours minimum)**

Communications General Education Requirement - Minimum 42 hours of (C) courses, at least 15 in major courses and at least 15 in non-major courses. A (C) designates courses used for Comm. Gen. Ed. Req.

**Interprofessional Projects**

<table>
<thead>
<tr>
<th>Course</th>
<th>Hrs.</th>
<th>Semester</th>
</tr>
</thead>
<tbody>
<tr>
<td>IPROxxx</td>
<td>(C) 3</td>
<td></td>
</tr>
<tr>
<td>IPROxxx</td>
<td>(C) 3</td>
<td></td>
</tr>
</tbody>
</table>

**Total hours (6 hours minimum)**

**Free Electives**

<table>
<thead>
<tr>
<th>Course</th>
<th>Hrs.</th>
<th>Semester</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>3</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3</td>
<td></td>
</tr>
</tbody>
</table>

**Total hours (12 hours minimum)**

**Minor field ____________________________**

A minor is *optional*. If a minor is selected, it must include a minimum of 15 hours. Place an M next to the courses used toward the minor.
Notes on the B.S. in Computer Science for Students Starting Fall 2004 thru Current

General note: Courses marked in the IIT Bulletin as not applying to graduation for degrees in "engineering and the physical sciences" may not be used toward the B.S. in Computer Science - this includes their use as free electives.

Computer science electives (5): Any computer science course at the 300-level or higher (including Graduate CS courses) may be used as a computer science elective, except CS 401, CS 402, CS 403 and CS 406. ECE 218(Digital Systems), ECE 441(Microcomputers), and ECE449(Object Oriented Programming and Computer Simulation) may also be used as computer science electives. To fulfill the university Communications General Education requirement, at least one CS elective must be chosen from CS422, CS425, CS429, CS447, CS470, CS471. No courses from any other programs can be used as computer science electives (for example, ICOM, ITM or CSP courses cannot be used as CS electives).

Mathematics elective (1): Must be chosen from MATH252, MATH410, MATH453, MATH454, MATH476, MATH482.

Science electives (2, no lab required): Chosen from the natural sciences (Biology, Chemistry, and Physics), or MS201-Materials Science, or Psychology (limited to courses marked with an N in the IIT Bulletin). At least one course must be in a field other than Physics.

Humanities and Social or Behavioral Sciences (7): 21 credit hours, subject to minimum requirements in each area as specified below:
Humanities: a minimum of nine credit hours. Courses that satisfy this requirement are marked with an (H) in the UG bulletin or in the WebForStudents Course Descriptions (Art & Architecture History, Literature, History, Humanities, Philosophy, Communications). The courses must be distributed as follows:
(a) Humanities 100-level course.
(b) At least two courses marked with an (H) at the 300 level or above. Some students may use foreign language courses at the 200 level to fulfill 300-level requirements. Students wishing to use foreign language courses must confirm their eligibility with the academic associate dean.

Social or Behavioral Sciences: a minimum of nine credit hours. Courses that satisfy this requirement are marked with an (S) in the UG bulletin or in the WebForStudents Course Descriptions (Anthropology, Economics, Political Science, Psychology, Sociology). The courses must be distributed as follows:
(a) At least two courses on the 300 level or above.
(b) Courses from at least two different fields.
(c) At least six credits in a single field.

Minor (Optional): A minor may be chosen from the specialized minors listed in the IIT Bulletin or may be formed from 15 hours of course work in one department. The latter option requires written approval from both the student's faculty advisor and the minor department.

ROTC: ROTC programs are considered to be minors and satisfy the requirements for minors listed above.

Graduate and short courses: Undergraduates may enroll in a 500-level graduate course with permission from the student's faculty advisor. Undergraduates cannot enroll in short courses.

Communications General Education Requirement: Minimum 42 hours of (C) courses as marked in UG bulletin or in the WebForStudents Course Descriptions, at least 15 hours in major courses and at least 15 hours in non-major courses. Almost all Humanities and Social or Behavioral Science electives will count towards the communications requirement.
## B.S. in Computer Science Sample Curriculum by Semester for Students Starting Fall 2004 thru Current

<table>
<thead>
<tr>
<th>First semester</th>
<th>Second semester</th>
<th>Third semester</th>
<th>Fourth semester</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS 100 2</td>
<td>CS 116 2</td>
<td>CS 331 3</td>
<td>CS 351 3</td>
</tr>
<tr>
<td>CS 115 2</td>
<td>CS 330 3</td>
<td>CS 350 3</td>
<td>CS 430 3</td>
</tr>
<tr>
<td>MATH 151 5</td>
<td>MATH 152 5</td>
<td>MATH 251 4</td>
<td>MATH 332 or 333 3</td>
</tr>
<tr>
<td>HUM 100-level elective 3</td>
<td>PHYS 123 4</td>
<td>PHYS 221 4</td>
<td>Science elective 3</td>
</tr>
<tr>
<td>Social Science elective 3</td>
<td>Humanities elective 3</td>
<td>Social Science elective 3</td>
<td>Humanities elective 3</td>
</tr>
<tr>
<td></td>
<td>15</td>
<td>17</td>
<td>17</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Fifth semester</th>
<th>Sixth semester</th>
<th>Seventh semester</th>
<th>Eighth semester</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS 440 3</td>
<td>CS 450 3</td>
<td>CS 487 3</td>
<td>CS 485 3</td>
</tr>
<tr>
<td>CS elective 3</td>
<td>CS elective 3</td>
<td>CS elective 3</td>
<td>CS elective 3</td>
</tr>
<tr>
<td>MATH 474 or 475 3</td>
<td>Math elective 3</td>
<td>Science elective 3</td>
<td>Free elective 3</td>
</tr>
<tr>
<td>Social Science elective 3</td>
<td>IPRO I 3</td>
<td>IPRO II 3</td>
<td>Free elective 3</td>
</tr>
<tr>
<td>COM421 or 428 3</td>
<td>Free elective 3</td>
<td>Hum or Soc Sci elective 3</td>
<td>Free elective 3</td>
</tr>
<tr>
<td></td>
<td>15</td>
<td>15</td>
<td>15</td>
</tr>
</tbody>
</table>

Total credit hours 127
Official Academic Audit (sample)

December 5, 2007

Mr. B...

Dear Mr. B...

Your academic record indicates that you must complete the following courses:

<table>
<thead>
<tr>
<th>Course</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hum Elect (3001)</td>
<td>-3</td>
</tr>
<tr>
<td>Soc Sci Elect</td>
<td>-3</td>
</tr>
<tr>
<td>Soc Sci Elect (3001)</td>
<td>-3</td>
</tr>
<tr>
<td>Hum or Soc Sci Elect</td>
<td>-3</td>
</tr>
<tr>
<td>IPRO Elects</td>
<td>-6</td>
</tr>
<tr>
<td>Math 114 or 175</td>
<td>-3</td>
</tr>
<tr>
<td>Math Elects</td>
<td>-3</td>
</tr>
<tr>
<td>Phys 123</td>
<td>-4</td>
</tr>
<tr>
<td>Phys 221</td>
<td>-4</td>
</tr>
<tr>
<td>Science Elects</td>
<td>-6</td>
</tr>
<tr>
<td>CS 485</td>
<td>-3</td>
</tr>
<tr>
<td>CS 487</td>
<td>-3</td>
</tr>
<tr>
<td>CS Elects</td>
<td>-12</td>
</tr>
<tr>
<td>Comp 421 or 428</td>
<td>-3</td>
</tr>
<tr>
<td>Free Elects</td>
<td>-6</td>
</tr>
<tr>
<td>TOTAL</td>
<td>-65 hrs.</td>
</tr>
</tbody>
</table>

Note 1) Of the three Social Science courses required for graduation, two must be at the 3001 level. Courses must be chosen from two different fields and two courses must be in the same field.

Note 2) Please consult the 2004-07 undergraduate bulletin concerning the communication requirements.

To be eligible for graduation, a student must file an "Application for Graduation" at the beginning of the semester in which he or she plans to graduate and satisfy all Institute policies and regulations. Refer to current Bulletin for specific filing deadlines.

<table>
<thead>
<tr>
<th>Degree</th>
<th>Bachelor of Science</th>
</tr>
</thead>
<tbody>
<tr>
<td>Major</td>
<td>Computer Science</td>
</tr>
<tr>
<td>Specialization</td>
<td>--</td>
</tr>
<tr>
<td>Minor/Spec. Minor</td>
<td>--</td>
</tr>
<tr>
<td>Cumulative GPA</td>
<td>3.828</td>
</tr>
<tr>
<td>Major GPA</td>
<td>3.769</td>
</tr>
<tr>
<td>Audit inclusive of</td>
<td>2007 Summer</td>
</tr>
<tr>
<td>To be completed</td>
<td>65</td>
</tr>
<tr>
<td>Total</td>
<td>127</td>
</tr>
</tbody>
</table>

Illinois Institute of Technology
Office of Educational Services
3300 South Federal Street
Chicago, Illinois 60616
Telephone: 312-567-3300

Appendix II - 8
Undergraduate Advising Hints

http://www.cs.iit.edu/programs/bachelors/advising.html

The Role Of The Advisor
The goal of student advising is to assist the student in:
Monitoring progress toward graduation by fulfilling all degree requirements.
Selecting courses that meet the student's individual goals and career objectives.
Ensuring that the student takes an appropriate, balanced load of technical and nontechnical courses each semester while meeting all course prerequisites.
Dealing with problems such as the need to drop a course, probation, etc.

The IIT Bulletin of Undergraduate Programs specifies the degree requirements for the Bachelor of Science in Computer Science and the Bachelor of Science in Computer Information Systems. The year that a student is admitted to IIT defines the binding program for that student as found in that year's Bulletin. Changes made to the bulletin after a student enrolls at IIT generally do not alter the student's program of study (unless the student wishes to follow the changes voluntarily).

Sample 'semester by semester' programs for both CS and CIS are given below. Note that this program is only a guideline. Each student works closely with an advisor to formulate a program of study that not only meets the student's goals in terms of content, but that also allows for experiences outside of the classroom -- internships, co-ops, and ROTC programs, for instance.

Program Study Checklists

For students who started in Fall 2004 or later:
Bachelor of Science in Computer Science (BSCS)
Bachelor of Science in Computer Information Systems (BSCIS)

For BSCS students who started:
Fall 2002 - Spring 2004
Fall 2001 - Spring 2002
Fall 1998 - Spring 2001

For BSCIS students who started:
Fall 2001 - Spring 2004
Fall 1998 - Spring 2001

Student Files And Program Study Checklists
When a student enrolls at IIT, a file folder is created for each student by the Office of Admissions. This folder contains initial information about the new student, including placement exam scores, high school information, and copies of Transfer Evaluation forms for transfer courses evaluated by the Office of Educational Services. When the student first enrolls at IIT the file is transferred to their department for their degree program. As the student continues at IIT, the file is updated on an advisor-maintained
The CS/CIS Program of Study Checklist forms aid the advisor and student in monitoring the student's progress toward the CS/CIS degree. It shows the CS/CIS curricula provides spaces to 'check off' the term each course was taken and the number of credit hours. Transfer courses are marked with 'XFER', and courses replaced by advanced placement credits should be marked with 'AP'. For example, a transfer student checklist might include:

MATH 151 Calculus I 4 AP
CS 100 Intro to Profession 2 XFER
CHEM 124 Gen Chemistry 4 F93

The Undergraduate File Drawers in the CS department contains blank copies of the CS/CIS Program of Study Checklist forms.

During registration and advising, it is often important to review a student's records so that plans for future courses can be carefully evaluated. While records are kept in each student's folder, they are not always up to date. Up to date records of courses taken can be obtained from SIS.

Registration Procedures
The majority of undergraduates (at least after freshman year) are fairly competent at following the correct program of study and picking appropriate classes each semester. Officially undergraduates still need advisor approval, but this is much easier with email and Web for Students. This is the current process:

Approximately 2 weeks before registration starts, the department sends an email to all undergraduate CS and CIS majors telling them to check the next semester's schedule of classes (which should be available on Web for Students), and to choose courses with sections, days and times.

Students are reminded to check prerequisites and to follow the degree requirements in the IIT Bulletin of Undergraduate Programs. In general, undergraduates are not allowed to register for IIT/V or Internet sections.

If students have completed (or after this semester have completed) 70 credit hours total, they should make sure they have an updated Academic Audit done by Educational Services (first floor of Main Building) ASAP. Students should use this audit to help pick their courses in all future semesters instead of using their CS/CIS Program of Study Checklist.

Students send me an email (Advisor should save the students' emails) with their ss# (student ID) and what courses they want to register for to their advisor, who quickly checks the choices (either by quickly looking at their previous courses taken in SIS or by
pulling their file and looking at their CS/CIS Program of Study Checklist or latest Academic Audit) and releases the advisor hold (SIS Screen 148) so the student can register using Web for Students (there may also be a financial hold so students may need to talk to Student Services to release that).

Later, when time permits, the advisor will more carefully check to make sure students have registered correctly, and will update the Program of Study Checklists (or Academic Audit) in the Student Files. If any mistakes are seen, the advisor emails the student with recommended changes and student will need to drop/add (advising hold needs re-releasing).

Advisors will also be available for one-on-one advising during regular office hours.

IPROs: See the IPRO website for information on IPROs, finding IPRO projects, and enrolling in IPROs.

The Student Information System (SIS) for Advisors

The Student Information System (SIS) is maintained by the Office of Administrative Information Systems (AIS) in 110 MB (x7-5071). SIS provides advisors with online access to student records. SIS can be accessed by telnet to 'alpha1.ais.iit.edu' (this may change at some point in the future) or from Web for Faculty/Web For Advisors (webfor.iit.edu).

SIS operates as a series of screens that are indexed by a 'screen number' (SCREEN) field in the upper left-hand corner of each screen and fields for course number (CRS), student ID (SID), and term (TERM) which must be entered depending on which screen is requested. You can skip between these fields using TAB.

To access the record of a student, enter into the SID field either the student ID (social security number with no hyphens) or the student's name in the form 'lastname (space) firstname' (if the name is ambiguous, a 'name search' screen will allow you to select which student you want).

To access the record of a course, enter into the CRS field a four character department code (dept. initials padded with blanks) followed by the course number and (if needed) the section number. Examples: 'ECE 211', 'CS 201', 'MATH474001', 'ECE 446001', 'CS 105071'.

To specify a particular course term, enter into the TERM field a 2-digit year followed by a term specifier S (Spring), F (Fall), or M (Summer). Examples: '00F', '01M', '02S'.

The following screens are most useful during the advising process:

Screen 101: Menu of available screens.
Screen 104: Permit (P) a student for a course over the enrollment limit or waive pre-requisites (U) for a course for a student.
Screen 105: Course Sections -- lists sections of a course and summarizes enrollment in a given term. Useful for checking whether a course section is full.
Screen 107: Course Roster -- lists students in a given course section in a given term.
Screen 108: Wait List -- lists students waiting to get into a course.
Screen 109: Student Schedule -- lists student's schedule in a given term.
Screen 120: Visual Student Schedule -- lists student's schedule in a given term.
Screen 121: Academic Program Summary -- lists information about the student's program.
Screen 130: Course Section Meeting Schedule -- shows when a course's sections are meeting.
Screen 136: Student Transcript -- lists courses, grades, and transfer credit to date.
Screen 148: Release advisor holds and view all holds.
APPENDIX III – Assessment Materials

IIT Computer Science Undergraduate Program Evaluation Process

http://www.cs.iit.edu/~abet
http://www.cs.iit.edu/announcements/ugradEval.html

Undergraduate Program Evaluation is an ongoing, overlapping process, with assessments being done during Year #2 as we work on implementing Year #1 changes.

Year #1, Spring Semester
Student Course Assessment (end of every semester)
Instructor Course Assessment (end of every semester)
CS Undergraduate Committee Detailed Course Review (one course every semester).

Year #1, Summer Semester
Student Course Assessment (end of every semester)
Instructor Course Assessment (end of every semester)
Recent Alumni Survey
Co-Op/Employer Survey

Year #1, Fall Semester
Student Course Assessment (end of every semester)
Instructor Course Assessment (end of every semester)
CS Undergraduate Committee Detailed Course Review (one course every semester)
CS Advisory Committee Feedback
ACM Student Feedback Forum

Year #1, Winter
Summarize all assessments for the calendar year.
January: Distribute all summaries to Course Managers and Instructors, with required feedback.

Year #2, Spring Semester
By May: CS Undergraduate Committee reviews all Course Manager and Instructor feedback and makes recommendations for implementation.

Year #2, Summer Semester
Course Managers and Instructors prepare changes.

Year #2, Fall Semester
Course Managers and Instructors implement changes.

All original assessment documents may be found in the Computer Science Department, 236 Stuart Building or by contacting Matthew Bauer matthew.bauer@iit.edu ex.75148.
TO: Faculty Teaching Undergraduate Computer Science Courses (Math230 included)
FROM: Computer Science Undergraduate Committee
DATE: April 14, 2008
SUBJECT: Computer Science Undergraduate Course Student Assessments

As part of ABET Accreditation and ongoing improvement to the Undergraduate Computer Science courses, the Computer Science Department requires Undergraduate Course Student Assessments to be done at the end of every semester. The assessments are to be completed during the last 2 weeks of the semester. The Computer Science Department expects 100% of its courses to be assessed by our students. If possible, announce one class in advance that the assessment will be conducted.
Every different lecture section of a course should be assessed separately.
The Course Manager of the course and you, the instructor of the course, will review these assessments AFTER course grades have been recorded.

Please read and follow the instructions below prior to initiating the assessment procedure for your class.
Assessments are to be completed at the end of a single class period. No less than 15 minutes prior to the scheduled end of the class period, conclude class activities and begin the assessment process.
Read the following statement to the class: “Please complete the assessment form to be passed out in order to provide feedback about this course section. We would like you to be thoughtful in your responses and to provide responsible, objective answers to the questions. These data and comments will be reviewed by the Computer Science Department and me, the instructor, after grades have been submitted.”
Request a volunteer to distribute the assessments, collect the assessments and seal them in this envelop, and drop the sealed envelop of assessments off in the Computer Science Department (room 236 Stuart Building).
Immediately leave the classroom
LATER, PLEASE VERIFY WITH MATTHEW BAUER at matthew.bauer@iit.edu THAT YOUR STUDENT ASSESSMENTS WERE RECEIVED BY THE CS DEPT.
For courses with off-site students (IIT/V or internet sections) please attempt to send the “CS Course Student Assessment” to all off-site students via the IIT/V courier or via email (see www.cs.iit.edu/~abet for electronic copies of the assessment). Ask them to please return the completed assessment to the Computer Science Department (room 236 Stuart Building).
CS Course Student Assessment  
(similar for every undergraduate course offered)

Course ___CS105___  Semester ___SPRING 2008___  Grad or Undergrad (circle one)  
INSTRUCTOR: __________________________  SECTION: __________________

The CS department is committed to continuous improvement of its programs and would like to have more information than is provided by the standard IIT course evaluation form. Please take a few minutes to complete this survey. Thank you.

Did you understand what was expected of you in the course?  Yes / No

Do you think that you achieved the course goals (see back) for the course? Yes / No

If you answered “No” to question 2, which goal or goals did you not achieve (please list by number from back).
___________________________________________________________________________
___________________________________________________________________________
___________________________________________________________________________

Were you adequately prepared to take this course by your mathematics background?   Yes / No

Were you adequately prepared to take this course by prerequisite computer science courses (see back)?  
                       Yes / No / Not applicable

What did you like best about this course?
___________________________________________________________________________
___________________________________________________________________________
___________________________________________________________________________
___________________________________________________________________________

What, if anything, would you change about this course?
___________________________________________________________________________
___________________________________________________________________________
___________________________________________________________________________
___________________________________________________________________________

Please continue on the back if necessary.
CS105 - Introduction to Computer Programming I

Course Manager - Matthew Bauer, Senior Lecturer
2 credit hours; required for CS & CPE (or CS200); 100 min. lecture & 50 min. lab each week

Current Catalog Description - Introduces the use of a high-level programming language (C/C++) as a problem-solving tool—including basic data structures and algorithms, structured programming techniques, and software documentation. Designed for students who have had little or no prior experience with computer programming. (2-1-2)

Course Goals - Students should be able to:
Analyze and explain the behavior of simple programs involving the following fundamental programming constructs: assignment, I/O (including file I/O), selection, iteration, functions
Write a program that uses each of the following fundamental programming constructs: assignment, I/O (including file I/O), selection, iteration, functions
Break a problem into logical pieces that can be solved (programmed) independently.
Develop, and analyze, algorithms for solving simple problems.
Use a suitable programming language, and development environment, to implement, test, and debug algorithms for solving simple problems.
Write programs that use each of the following data structures (and describe how they are represented in memory): strings, arrays, and class libraries including strings and vectors
CS Faculty/Course Self-Assessment

(similar for every undergraduate course offered)

Course: CS105  Semester: SPRING 2008  # of students (live): _____  (remote): _____

INSTRUCTOR: __________________________  SECTIONS: __________________

1. How many students earned each grade?
   (live)      A ______  B ______  C ______  D ______  E ______
   (remote) A ______  B ______  C ______  D ______  E ______

2. Indicate how many students met each course goal by placing an X in each row for each Course Goal.

<table>
<thead>
<tr>
<th>COURSE GOALS</th>
<th>ALL</th>
<th>MOST</th>
<th>SOME</th>
<th>NONE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Analyze and explain the behavior of simple programs involving the following fundamental programming constructs: assignment, I/O (including file I/O), selection, iteration, functions</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Write a program that uses each of the following fundamental programming constructs: assignment, I/O (including file I/O), selection, iteration, functions</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Break a problem into logical pieces that can be solved (programmed) independently.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Develop, and analyze, algorithms for solving simple problems.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Use a suitable programming language, and development environment, to implement, test, and debug algorithms for solving simple problems.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Write programs that use each of the following data structures (and describe how they are represented in memory): strings, arrays, and class libraries including strings and vectors</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

3. In your opinion, were the students adequately prepared in mathematics? (Yes/No). If "No", what deficiencies did you identify? ________________________________________________________________
   __________________________________________________________________________
   __________________________________________________________________________

4. In your opinion, were the students adequately prepared in computer science basics? (Yes/No). If "No", what deficiencies did you identify? ________________________________________________________________
   __________________________________________________________________________
   __________________________________________________________________________

5. In your opinion, were the students adequately prepared by other pre-requisite courses, if any? (Yes/No). If "No", what deficiencies did you identify? ________________________________________________________________
   __________________________________________________________________________
   __________________________________________________________________________

6. If you were to teach the course again, what changes would you make? (Use additional pages if needed)
   __________________________________________________________________________
   __________________________________________________________________________
   __________________________________________________________________________

   Appendix III - 5
Communication Across the Curriculum Assessment 2006

This document includes a description of the current program and a description of assessment measures and results.

Description of the Current Program
Communication Across the Curriculum (CAC) is a faculty-directed and faculty-approved curricular program in writing and speaking skills whose responsibilities include instruction and assessment:

Instruction
- Teaching students (primarily undergraduates) about writing and speaking skills
- Helping teachers in each discipline to develop effective communication-oriented assignments
- Assessment
- Monitoring discipline-based instruction about writing and speaking skills in three kinds of course: Introduction to the Profession (ITPs), Communication-intensive courses in all disciplines (C-courses), and Interprofessional Projects (IPROs and ENPROs)
- Evaluating the overall effectiveness of communication instruction through ongoing longitudinal studies in cooperation with the IPRO program and the Office of Educational Services
- Evaluating students’ admission essays in order to determine appropriate placement into specialized writing courses

To accomplish its instructional tasks, the CAC program integrates a variety of instructional modes into a single, coherent, system of mutually reinforcing parts, serving students in different ways to address ascending levels of need:
- At the lowest level of need, online webpages that offer specific guidelines and techniques for communication assignments in ITPs and C-courses developed in consultation with course instructors, and IPROs, along with general instruction about writing
- At a higher level of need, tutors provide individual feedback about writing assignments in the CAC Writing Center and the Humanities Writing Center
- At the highest level of need, students are required to pass an introductory or advanced writing courses (COM 101/111, Introduction to College Writing; COM 421, Technical Writing)

In other words, students with relatively slight deficiencies can take advantage of course-specific webpages, while students with more serious deficiencies can seek help at the IIT Writing Center (formerly the separate Humanities and CAC writing centers), and students with very significant problems can be referred to writing courses.
The CAC also coordinates the development and delivery of instructional materials with the Technical Communication Program (Humanities Department) and individual instructors ITPs, C-courses, and IPROs.

To accomplish its assessment tasks, the CAC program collaborates with several university units:

- With the former Humanities Writing Center, the CAC offers 40 hour of tutoring in writing and speaking assignments each week in a unified IIT Writing Center, including specialized assistance for nonnative speakers of English.
- With the Office of Educational Services, the CAC collects and analyzes data (GPA, retention) to develop periodic reports on the use and effectiveness of instructional resources and the overall program.
- With the Undergraduate Studies Committee, the CAC will consult with a USC subcommittee to review communication instruction and assessment on a continuing basis (beginning in Fall 2006).
- With the Interprofessional Projects program, the CAC collects and analyzes data relevant to writing and speaking tasks required in every IPRO, including oral presentations and poster sessions on each semester’s IPRO Day.
- With the Digital Media Center, the CAC is developing instructional videos and feedback mechanisms for students.

In cooperation with IIT’s Digital Media Center, the CAC plans to develop an instructional module on oral presentations. This module will include an instructional DVD with a series of brief presentations by Dr. Greg Pulliam (already completed), as well as a video area in the Digital Media Center that will allow students to practice oral presentations, view and critique themselves, and receive feedback from a CAC tutor. Feedback norms and forms have already been developed.

IIT is currently seeking a new director for the CAC program.

Assessment Measures and Results

1. External Assessment by ABET
   The Communication Across the Curriculum program was last assessed by an external agency during the ABET review in 2003, whose evaluators characterized the CAC program as an “institutional strength” and a “hallmark of the institution” (along with the IPRO program).

2. Placement Activities for New Students
   CAC staff evaluate writing samples of incoming students to assess their Basic Writing Proficiency. An earlier placement portfolio for this purpose consisted of diverse genres (including poetry and plays) not clearly relevant to student writing tasks. An internal review undertaken in preparation for the ABET evaluation had shown that this approach was not yielding acceptable results. Therefore, a new approach was installed, including a writing sample with new set of prompts.
(instructions) designed to evoke essays based on typical academic tasks (e.g., describing a process, describing a situation and its effects).

Results of scoring patterns by separate evaluators have been monitored for five years. Pairs of essay evaluators show a high degree of inter-reader reliability, with significantly disparate scores occurring in less than 10% of the cases. Of the four essay prompts (directions to students), three were found to yield consistent results over a three-year period; but a fourth prompt yielded consistently lower scores, and was dropped. Evaluations are consistent over time, with approximately 25% of entering students being required to take COM 101 or 111 each year. As of 2005-2006, essay scores are being informally correlated with ACT and SAT scores, so that recommendations are based on multiple measures. This is resulting in very few challenges of test results.

3. Longitudinal Study of Writing Success
In 2002-2003, the CAC began a long-range, continuing evaluation of placement as well as the overall success of the instructional.

In an initial report to be completed in Fall 2006, placement scores will be matched with enrollment patterns to measure retention (i.e., the effectiveness of placement-required instruction on a student’s persistence as a student at IIT). Placement scores, grades in COM 101 or 111, and Writing Center usage will be matched with overall GPA, GPA in communication-intensive courses (C-courses), GPA in Introduction to the Profession (ITP) courses, semester-by-semester retention rates, and successful graduation.

4. IIT Writing Center
Previously, two writing centers were maintained (one as part of the Academic Resource Center and one as part of the Humanities Department). To centralize tutoring resources, the CAC facility began to share the same space as the Humanities Writing Center in 2003-2004, resulting in consistent instructional materials being presented to students in tutoring sessions. In 2005-2006, the two tutorial programs merged their staff and resources, with the Humanities Writing Center director (James Dabbert) serving as a volunteer associate director of the CAC tutoring effort, with responsibility for supervising the day-to-day activities of all Writing Center staff. During this period of consolidation, student usage of the writing center has increased steadily. In 2006-2007, Dabbert will begin to receive compensation for official service as the associate director of the CAC, with responsibility for managing all Writing Center staff and services. Compensation was set at $10,000 for 2006-2007 by approval of Interim Undergraduate College Dean Donald Ucci, with the approval of Provost Myerson. In 2006-2007, Dabbert will also serve as Interim Director of the CAC.

Writing Center usage in 2005-2006 is included in the report for the Humanities Department, which now houses the CAC program.

5. Instructional Materials for CAC Website and in IIT Writing Center
As part of their general education curriculum, students must take 42 credit-hours of writing-intensive courses. To assist instructors and students in these courses,
the CAC maintains a website that links a course instructor’s directions and guidelines with supplementary guidelines consistent with instruction in IIT writing courses, including COM 101 (University Writing) and COM 421 (Technical Writing). In 2005-2006, a content analysis of these instructional materials was conducted. Prospective employers and employees in four IIT degree programs were interviewed, along with IIT instructors and students in the programs. As results showed, the skills emphasized in CAC instructional materials and in supportive courses such as COM 421, “Technical Writing” (required for degrees in the Computer Science department), and COM 423, “Writing in the Science Workplace” (required for several degree programs in the Biology, Chemistry, and Physical Science department), receive a high priority among future employers and fellow employees of IIT students.

6. Professional Training for Writing Center Tutors
Starting in 2004-2005, tutors in the writing center were required to complete COM 561, “Teaching Technical Communication,” and COM 535, “Instructional Design.” Starting in 2006-2007, tutors will also be required to take a new communication course specifically designed for the writing center: COM 562, “Conferencing and Assessment.”

7. Surveys of Former Students
According to a 2006 EBI Engineering Alumni Assessment (#10658), IIT engineering alumni felt that they had been successfully prepared for workplace writing tasks. Complete data are available online at this URL:


8. Programmatic Feedback through Academic Conference Presentations
In 2005-2006, CAC faculty and graduate assistants described IIT’s programmatic efforts and received positive feedback from presentations at five professional conferences, including the following:


“Analyzing Jobs and Assessing Programs.” Association of Teachers of Technical Writing Conference, in conjunction with Conference on College Composition and Communication (Chicago, IL, March 21-25, 2006). With Michael Tillmans and Julia Chase.

“Creating Course-Specific Instructional Websites about Writing Tasks.” Writing Program Administrators conference (Anchorage, AK, July 7-10, 2005).

“Integrating Real and Virtual Space in Communication Across the Curriculum.” College English Association conference (Indianapolis, IN, March 31-April 2, 2005). With Laura Batson, Don Cunningham, Divya Jindal, , Pat Johnson-Winston, and Jyothi Shankar.
Appendix A.  IIT Writing Center Report for Spring 2006

The IIT Writing Center is located on the main campus, Siegel Hall, Rooms 232 and 233 and open to all members of the academic community.

For the Spring 2006 semester, the Writing Center opened on January 30 and closed on May 12. This past semester marked the complete integration of the Humanities Writing Center and the Communication Across the Curriculum program, an initiative of Dr. Glenn Broadhead. Serving as tutors were Gail Lehman, Humanities adjunct faculty, Chris Lam and Mindy Sherman, graduate students in the Technical Writing and Communication Across the Curriculum programs, and James Dabbert, Senior Lecture in English, Writing Center director. Gail Lehman, who has served as tutor for the past three semesters, has informed the Department that she will not be returning for Fall 2006. We all wish her the best.

The Writing Center was opened five days a week, Monday through Friday, with a typical daily schedule of 9:30 AM to 6:00 pm. Weekly student sign-up sheets were posted on the faculty offices of 232 and 233, allowing visitors to make their own appointments. Most appointment lasted from one half to one hour.

The total number of Spring 2006 visitors was 151. This number represents the third highest attendance since I began keeping records in 1988. The other two semesters were Fall 1993 with 154 students, and Fall 1994 with 165 students. For Spring 2006, the number of undergraduate was 101 and 49 graduate students, and one visiting faculty member from Iran. Non-native speakers totaled 120, native English speakers 31. Total tutoring sessions numbered 336. Most students are self-referrals, with 63 referred by faculty members. Most popular were students from the College of Architecture, Electrical Engineering, and Computer Science, each department represented by 19 students each. Korean speakers made up the largest number of visitors at 41, with English a close second at 31.

The success of the IIT Writing Center depends on the effectiveness of our service to the faculty, and in this spirit, we welcome suggestions. The Humanities Writing Center will reopen on Sept 4, 2006.

James R. Dabbert
Director, Humanities Writing Center
Siegel Hall Room 233
Tel: 312 567-5188
dabbert@iit.edu

NOTE: Additional detailed data is available in the Writing Center Report included in the Humanities Department’s assessment report
Engineering Accreditation 2008
Evidence

Pages 1-15, Addendum A: Network Reports (pages 1-10)
Telecommunications Reports (pages 11-13)
IIT Online Reports (page 14)


Page 26, Addendum C: 2006-2008 Building Technology Infrastructure Upgrades

Pages 27-28, Addendum D: Classroom A/V Inventory List

Computer Science Master Software List 2008

Pages 31-37, Addendum F: IIT On-line Training Manual for Faculty

Page 38, Addendum G: Blackboard Training Sheet for Faculty
Total Monthly Incoming Messages (and % Filtered Spam)

No. of Email messages

% Filtered

Total Monthly Incoming Messages (and % Filtered Spam)

Total Messages

Messages Filtered

Feb-07  Mar-07  Apr-07  May-07  Jun-07  Jul-07  Aug-07  Sep-07  Oct-07  Nov-07  Dec-07  Jan-08  Feb-08

Appendix -IV Facilities
Number of PCs in Quarantine

Note - Email anti-virus tool implemented in April 2007
Percentage of Email accounts above 80% Quota for Students/Faculty/Staff

<table>
<thead>
<tr>
<th>Months</th>
<th>Number of accounts</th>
<th>Percentage above 80%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Feb-07</td>
<td>17046</td>
<td>2.21%</td>
</tr>
<tr>
<td>Mar-07</td>
<td>17114</td>
<td>2.69%</td>
</tr>
<tr>
<td>Apr-07</td>
<td>21214</td>
<td>2.75%</td>
</tr>
<tr>
<td>May-07</td>
<td>21884</td>
<td>2.71%</td>
</tr>
<tr>
<td>Jun-07</td>
<td>22319</td>
<td>2.95%</td>
</tr>
<tr>
<td>Jul-07</td>
<td>22609</td>
<td>3.06%</td>
</tr>
<tr>
<td>Aug-07</td>
<td>23514</td>
<td>3.33%</td>
</tr>
<tr>
<td>Sep-07</td>
<td>23891</td>
<td>3.76%</td>
</tr>
<tr>
<td>Oct-07</td>
<td>24163</td>
<td>3.85%</td>
</tr>
<tr>
<td>Nov-07</td>
<td>14845</td>
<td>7.01%</td>
</tr>
<tr>
<td>Dec-07</td>
<td>14941</td>
<td>7.22%</td>
</tr>
<tr>
<td>Jan-08</td>
<td>15221</td>
<td>7.04%</td>
</tr>
<tr>
<td>Feb-08</td>
<td>15362</td>
<td>7.34%</td>
</tr>
</tbody>
</table>

Appendix -IV Facilities
Number of Internet Outages

- Planned
- Unplanned

<table>
<thead>
<tr>
<th>Months</th>
<th>No. of Outages</th>
</tr>
</thead>
<tbody>
<tr>
<td>Feb-07</td>
<td>0</td>
</tr>
<tr>
<td>Mar-07</td>
<td>1</td>
</tr>
<tr>
<td>Apr-07</td>
<td>0</td>
</tr>
<tr>
<td>May-07</td>
<td>0</td>
</tr>
<tr>
<td>Jun-07</td>
<td>0</td>
</tr>
<tr>
<td>Jul-07</td>
<td>0</td>
</tr>
<tr>
<td>Aug-07</td>
<td>0</td>
</tr>
<tr>
<td>Sep-07</td>
<td>1</td>
</tr>
<tr>
<td>Oct-07</td>
<td>0</td>
</tr>
<tr>
<td>Nov-07</td>
<td>0</td>
</tr>
<tr>
<td>Dec-07</td>
<td>0</td>
</tr>
<tr>
<td>Jan-08</td>
<td>0</td>
</tr>
<tr>
<td>Feb-08</td>
<td>0</td>
</tr>
</tbody>
</table>

Appendix -IV Facilities
Duration of Internet Outages by Cause

Appendix - IV Facilities
Unplanned Non-Internet Incidents/Outages by Location

Appendix -IV Facilities
Number of Times that 90% of Bandwidth is Reached

- ICN/RCN - 119 Mbps
- ICN/RCN - 165 Mbps
- ICN/COGENT - 215 Mbps

Appendix - IV Facilities
Total Duration that 90% of Bandwidth is Reached

ICN/RCN - 119 Mbps
ICN/RCN - 165 Mbps
ICN/COGENT - 215 Mbps

Note - % is Percentage of month
Maximum/Average Internet2 Bandwidth Usage

<table>
<thead>
<tr>
<th>Month</th>
<th>Max Mbps</th>
<th>Avg Mbps</th>
</tr>
</thead>
<tbody>
<tr>
<td>Feb-07</td>
<td>74</td>
<td>54</td>
</tr>
<tr>
<td>Mar-07</td>
<td>65</td>
<td>54</td>
</tr>
<tr>
<td>Apr-07</td>
<td>54</td>
<td>45</td>
</tr>
<tr>
<td>May-07</td>
<td>60</td>
<td>63</td>
</tr>
<tr>
<td>Jun-07</td>
<td>56</td>
<td>45</td>
</tr>
<tr>
<td>Jul-07</td>
<td>102</td>
<td>67</td>
</tr>
<tr>
<td>Aug-07</td>
<td>41</td>
<td>31</td>
</tr>
<tr>
<td>Sep-07</td>
<td>52</td>
<td>7</td>
</tr>
<tr>
<td>Oct-07</td>
<td>31</td>
<td>5</td>
</tr>
<tr>
<td>Nov-07</td>
<td>67</td>
<td>5</td>
</tr>
<tr>
<td>Dec-07</td>
<td>52</td>
<td>5</td>
</tr>
<tr>
<td>Jan-08</td>
<td>45</td>
<td>5</td>
</tr>
<tr>
<td>Feb-08</td>
<td>63</td>
<td>7</td>
</tr>
</tbody>
</table>

Appendix - IV Facilities
Posting problems – one course not posted on-time
Playback Problems – 10 tickets on streaming issues/slider issues. Investigating with Network Services. Possible issues with switch in servers temporarily with storage disk array swap and swap by network from Stuart/Perlstein
Other problems – Noise from another class, camera issue
Zoomerang Survey Results

Response Status: Completes
Filter: No filter applied
Apr 07, 2008 6:06 AM PST

1. Please select the campus where you most frequently attend classes.

<table>
<thead>
<tr>
<th>Campus</th>
<th>Responses</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Main Campus</td>
<td>1059</td>
<td>95%</td>
</tr>
<tr>
<td>Downtown Campus - Stuart School of Business</td>
<td>56</td>
<td>5%</td>
</tr>
<tr>
<td>Total</td>
<td>1115</td>
<td>100%</td>
</tr>
</tbody>
</table>

2. Please select the OTS Computer Labs you have used in the past year:

<table>
<thead>
<tr>
<th>Lab Name</th>
<th>Responses</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>3410 State Building, room 110</td>
<td>155</td>
<td>14%</td>
</tr>
<tr>
<td>Alumni Memorial, room 218</td>
<td>159</td>
<td>14%</td>
</tr>
<tr>
<td>Commons Night Owl Lab</td>
<td>174</td>
<td>15%</td>
</tr>
<tr>
<td>Crown Hall, room 001</td>
<td>207</td>
<td>18%</td>
</tr>
<tr>
<td>Downtown Campus Stuart Lab, room 180</td>
<td>40</td>
<td>4%</td>
</tr>
<tr>
<td>Downtown Campus Stuart Lab, room 401 (QRL Lab)</td>
<td>41</td>
<td>4%</td>
</tr>
<tr>
<td>Downtown Campus Stuart Lab, room 405</td>
<td>48</td>
<td>4%</td>
</tr>
<tr>
<td>Engineering One, room 029</td>
<td>419</td>
<td>37%</td>
</tr>
<tr>
<td>Hermann Union Building (HUB), room 112</td>
<td>226</td>
<td>20%</td>
</tr>
<tr>
<td>Residence Hall East, room 001</td>
<td>259</td>
<td>23%</td>
</tr>
<tr>
<td>Siegel Hall, room 236</td>
<td>327</td>
<td>29%</td>
</tr>
<tr>
<td>Siegel Hall, room 237</td>
<td>295</td>
<td>26%</td>
</tr>
<tr>
<td>Stuart Building, room 112</td>
<td>636</td>
<td>56%</td>
</tr>
<tr>
<td>Rice Campus, room 207</td>
<td>15</td>
<td>1%</td>
</tr>
<tr>
<td>Rice Campus, room 210</td>
<td>12</td>
<td>1%</td>
</tr>
<tr>
<td>I Have Not Used Any OTS Computer Labs in the Past Year</td>
<td>75</td>
<td>7%</td>
</tr>
</tbody>
</table>

3. I am satisfied with the equipment in the OTS computer labs.

<table>
<thead>
<tr>
<th>Satisfaction Level</th>
<th>Responses</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Strongly Disagree</td>
<td>14</td>
<td>1%</td>
</tr>
</tbody>
</table>

Appendix - IV Facilities
| Disagree | 53 | 5% |
| Neutral | 142 | 13% |
| Agree | 629 | 60% |
| Strongly Agree | 216 | 20% |
| * N/A * | 2 | 0% |
| **Total** | 1056 | 100% |

### 4. I am satisfied with the number of PCs available in the OTS Computer Labs.

| Strongly Disagree | 14 | 1% |
| Disagree | 97 | 9% |
| Neutral | 136 | 13% |
| Agree | 517 | 49% |
| Strongly Agree | 289 | 27% |
| * N/A * | 2 | 0% |
| **Total** | 1055 | 100% |

### 5. The software in the OTS Computer Labs supports my academic needs

| Strongly Disagree | 9 | 1% |
| Disagree | 64 | 6% |
| Neutral | 120 | 11% |
| Agree | 483 | 46% |
| Strongly Agree | 374 | 36% |
| * N/A * | 3 | 0% |
| **Total** | 1053 | 100% |

### 6. I am satisfied with the hours of the OTS Computer Labs.

| Strongly Disagree | 27 | 3% |
| Disagree | 146 | 14% |
| Neutral | 196 | 19% |
| Agree | 480 | 46% |
| Strongly Agree | 193 | 18% |
| * N/A * | 12 | 1% |

Appendix -IV Facilities
8. Please select how often you go to OTS labs for each of the following services:

<table>
<thead>
<tr>
<th>Service</th>
<th>Frequently</th>
<th>Sometimes</th>
<th>Rarely</th>
<th>Never</th>
</tr>
</thead>
<tbody>
<tr>
<td>Access to Computers</td>
<td>340</td>
<td>414</td>
<td>227</td>
<td>74</td>
</tr>
<tr>
<td></td>
<td>32%</td>
<td>39%</td>
<td>22%</td>
<td>7%</td>
</tr>
<tr>
<td>Specialized Software</td>
<td>278</td>
<td>410</td>
<td>258</td>
<td>99</td>
</tr>
<tr>
<td></td>
<td>27%</td>
<td>39%</td>
<td>25%</td>
<td>9%</td>
</tr>
<tr>
<td>Printing and Scanning</td>
<td>584</td>
<td>296</td>
<td>127</td>
<td>51</td>
</tr>
<tr>
<td></td>
<td>55%</td>
<td>28%</td>
<td>12%</td>
<td>5%</td>
</tr>
<tr>
<td>Study/Meeting Space</td>
<td>151</td>
<td>291</td>
<td>373</td>
<td>233</td>
</tr>
<tr>
<td></td>
<td>14%</td>
<td>28%</td>
<td>36%</td>
<td>22%</td>
</tr>
</tbody>
</table>

9. I understand how to print from my personal desktop/laptop to a release station.

| Strongly Disagree | 179 | 16% |
| Disagree          | 261 | 23% |
| Neutral           | 131 | 12% |
| Agree             | 229 | 20% |
| Strongly Agree    | 165 | 15% |
| Don't know        | 166 | 15% |
| Total             | 1131| 100%|
10. The number of usernames and passwords I have for IIT computing accounts is manageable.

<table>
<thead>
<tr>
<th>Rating</th>
<th>Responses</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Strongly Disagree</td>
<td>36</td>
<td>3%</td>
</tr>
<tr>
<td>Disagree</td>
<td>57</td>
<td>5%</td>
</tr>
<tr>
<td>Neutral</td>
<td>140</td>
<td>12%</td>
</tr>
<tr>
<td>Agree</td>
<td>572</td>
<td>51%</td>
</tr>
<tr>
<td>Strongly Agree</td>
<td>315</td>
<td>28%</td>
</tr>
<tr>
<td>* N/A *</td>
<td>9</td>
<td>1%</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td>1129</td>
<td>100%</td>
</tr>
</tbody>
</table>

11. The myIIT portal has improved my access to IIT information systems, news, weblinks, tools and other university content.

<table>
<thead>
<tr>
<th>Rating</th>
<th>Responses</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Strongly Disagree</td>
<td>54</td>
<td>5%</td>
</tr>
<tr>
<td>Disagree</td>
<td>78</td>
<td>7%</td>
</tr>
<tr>
<td>Neutral</td>
<td>201</td>
<td>18%</td>
</tr>
<tr>
<td>Agree</td>
<td>461</td>
<td>41%</td>
</tr>
<tr>
<td>Strongly Agree</td>
<td>310</td>
<td>28%</td>
</tr>
<tr>
<td><em>N/A</em></td>
<td>23</td>
<td>2%</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td>1127</td>
<td>100%</td>
</tr>
</tbody>
</table>

12. I have used myIIT to (select all that apply):

<table>
<thead>
<tr>
<th>Feature</th>
<th>Responses</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Check IIT Email</td>
<td>868</td>
<td>79%</td>
</tr>
<tr>
<td>Access Web for Students</td>
<td>963</td>
<td>88%</td>
</tr>
<tr>
<td>Go to Blackboard</td>
<td>934</td>
<td>85%</td>
</tr>
<tr>
<td>Join Groups</td>
<td>149</td>
<td>14%</td>
</tr>
<tr>
<td>Create a Calendar</td>
<td>97</td>
<td>9%</td>
</tr>
<tr>
<td>Learn about Campus Events</td>
<td>412</td>
<td>38%</td>
</tr>
<tr>
<td>Look at my Printing Account Balance</td>
<td>406</td>
<td>37%</td>
</tr>
<tr>
<td>Complete a Time Sheet for my Campus Job</td>
<td>309</td>
<td>28%</td>
</tr>
</tbody>
</table>

13. If there are any technology features you would like to see added or changed in myIIT, please provide them below:

Appendix -IV Facilities
14. Have you used the Blackboard course management system for any of your classes?

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Yes</td>
<td>950</td>
<td>84%</td>
</tr>
<tr>
<td>No</td>
<td>185</td>
<td>16%</td>
</tr>
<tr>
<td>Total</td>
<td>1135</td>
<td>100%</td>
</tr>
</tbody>
</table>

15. I am satisfied with the Blackboard system.

<table>
<thead>
<tr>
<th>Response Level</th>
<th>Count</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Strongly Disagree</td>
<td>18</td>
<td>2%</td>
</tr>
<tr>
<td>Disagree</td>
<td>41</td>
<td>4%</td>
</tr>
<tr>
<td>Neutral</td>
<td>134</td>
<td>14%</td>
</tr>
<tr>
<td>Agree</td>
<td>555</td>
<td>59%</td>
</tr>
<tr>
<td>Strongly Agree</td>
<td>198</td>
<td>21%</td>
</tr>
<tr>
<td>* N/A</td>
<td>1</td>
<td>0%</td>
</tr>
<tr>
<td>Total</td>
<td>947</td>
<td>100%</td>
</tr>
</tbody>
</table>

16. I would like to see more courses utilize the Blackboard system.

<table>
<thead>
<tr>
<th>Response Level</th>
<th>Count</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Strongly Disagree</td>
<td>13</td>
<td>1%</td>
</tr>
<tr>
<td>Disagree</td>
<td>29</td>
<td>3%</td>
</tr>
<tr>
<td>Neutral</td>
<td>139</td>
<td>15%</td>
</tr>
<tr>
<td>Agree</td>
<td>333</td>
<td>35%</td>
</tr>
<tr>
<td>Strongly Agree</td>
<td>420</td>
<td>44%</td>
</tr>
<tr>
<td>* N/A</td>
<td>11</td>
<td>1%</td>
</tr>
<tr>
<td>Total</td>
<td>945</td>
<td>100%</td>
</tr>
</tbody>
</table>

17. If there are any features you would like to see added or changed in Blackboard, please provide them below:

199 Responses
18. Please select how you utilize your IIT e-mail account:

<table>
<thead>
<tr>
<th>Method</th>
<th>Count</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Send/Receive e-mail through myIIT</td>
<td>708</td>
<td>62%</td>
</tr>
<tr>
<td>Send/Receive e-mail through another program (ex. Outlook, Eudora)</td>
<td>231</td>
<td>20%</td>
</tr>
<tr>
<td>Forward IIT e-mail to another e-mail address</td>
<td>455</td>
<td>40%</td>
</tr>
<tr>
<td>Do not check IIT e-mail</td>
<td>14</td>
<td>1%</td>
</tr>
<tr>
<td>Other, Please Specify</td>
<td>49</td>
<td>4%</td>
</tr>
</tbody>
</table>

19. If you use your IIT e-mail address, do you use it as your primary e-mail address for personal use?

<table>
<thead>
<tr>
<th>Response</th>
<th>Count</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yes</td>
<td>365</td>
<td>33%</td>
</tr>
<tr>
<td>No</td>
<td>726</td>
<td>65%</td>
</tr>
<tr>
<td>I Don't Use IIT e-mail</td>
<td>25</td>
<td>2%</td>
</tr>
<tr>
<td>Total</td>
<td>1116</td>
<td>100%</td>
</tr>
</tbody>
</table>

20. The storage space in my IIT e-mail mailbox meets my academic needs.

<table>
<thead>
<tr>
<th>Response</th>
<th>Count</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Strongly Disagree</td>
<td>142</td>
<td>13%</td>
</tr>
<tr>
<td>Disagree</td>
<td>174</td>
<td>16%</td>
</tr>
<tr>
<td>Neutral</td>
<td>209</td>
<td>19%</td>
</tr>
<tr>
<td>Agree</td>
<td>394</td>
<td>35%</td>
</tr>
<tr>
<td>Strongly Agree</td>
<td>161</td>
<td>14%</td>
</tr>
<tr>
<td>* N/A *</td>
<td>32</td>
<td>3%</td>
</tr>
<tr>
<td>Total</td>
<td>1112</td>
<td>100%</td>
</tr>
</tbody>
</table>

21. Please select the methods you use to connect to the IIT network:

<table>
<thead>
<tr>
<th>Method</th>
<th>Count</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wired network connection</td>
<td>516</td>
<td>46%</td>
</tr>
<tr>
<td>Wireless network connection</td>
<td>970</td>
<td>86%</td>
</tr>
<tr>
<td>Other, Please Specify</td>
<td>13</td>
<td>1%</td>
</tr>
</tbody>
</table>
### 22. I am satisfied with the speed of the IIT network.

<table>
<thead>
<tr>
<th>Strongly Disagree</th>
<th>94</th>
<th>8%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Disagree</td>
<td>165</td>
<td>15%</td>
</tr>
<tr>
<td>Neutral</td>
<td>246</td>
<td>22%</td>
</tr>
<tr>
<td>Agree</td>
<td>469</td>
<td>42%</td>
</tr>
<tr>
<td>Strongly Agree</td>
<td>151</td>
<td>13%</td>
</tr>
<tr>
<td>* N/A *</td>
<td>4</td>
<td>0%</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td>1129</td>
<td>100%</td>
</tr>
</tbody>
</table>

### 23. I am satisfied with the reliability of the IIT network.

<table>
<thead>
<tr>
<th>Strongly Disagree</th>
<th>94</th>
<th>8%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Disagree</td>
<td>194</td>
<td>17%</td>
</tr>
<tr>
<td>Neutral</td>
<td>266</td>
<td>24%</td>
</tr>
<tr>
<td>Agree</td>
<td>448</td>
<td>40%</td>
</tr>
<tr>
<td>Strongly Agree</td>
<td>117</td>
<td>10%</td>
</tr>
<tr>
<td>* N/A *</td>
<td>7</td>
<td>1%</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td>1126</td>
<td>100%</td>
</tr>
</tbody>
</table>

### 24. I feel there is sufficient wireless coverage on IIT’s Main Campus (and/or Stuart School of Business areas)

<table>
<thead>
<tr>
<th>Strongly Disagree</th>
<th>66</th>
<th>6%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Disagree</td>
<td>137</td>
<td>12%</td>
</tr>
<tr>
<td>Neutral</td>
<td>214</td>
<td>19%</td>
</tr>
<tr>
<td>Agree</td>
<td>457</td>
<td>41%</td>
</tr>
<tr>
<td>Strongly Agree</td>
<td>169</td>
<td>15%</td>
</tr>
<tr>
<td>* N/A *</td>
<td>81</td>
<td>7%</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td>1124</td>
<td>100%</td>
</tr>
</tbody>
</table>

### 25. I am aware of the IIT Computer Usage Policy.

<table>
<thead>
<tr>
<th>Yes</th>
<th>931</th>
<th>83%</th>
</tr>
</thead>
</table>

Appendix -IV Facilities
### 26. Please select which of the following residences you live in:

<table>
<thead>
<tr>
<th>Residence</th>
<th>Count</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>McCormick Student Village</td>
<td>254</td>
<td>22%</td>
</tr>
<tr>
<td>State Street Village</td>
<td>103</td>
<td>9%</td>
</tr>
<tr>
<td>IIT Graduate Apartments</td>
<td>95</td>
<td>8%</td>
</tr>
<tr>
<td>IIT Greek Housing</td>
<td>77</td>
<td>7%</td>
</tr>
<tr>
<td>None of the Above</td>
<td>606</td>
<td>53%</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td>1135</td>
<td>100%</td>
</tr>
</tbody>
</table>

### 27. The network in my residence is sufficient for my needs.

<table>
<thead>
<tr>
<th>Rating</th>
<th>Count</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Strongly Disagree</td>
<td>64</td>
<td>12%</td>
</tr>
<tr>
<td>Disagree</td>
<td>95</td>
<td>18%</td>
</tr>
<tr>
<td>Neutral</td>
<td>101</td>
<td>19%</td>
</tr>
<tr>
<td>Agree</td>
<td>200</td>
<td>38%</td>
</tr>
<tr>
<td>Strongly Agree</td>
<td>64</td>
<td>12%</td>
</tr>
<tr>
<td>* N/A *</td>
<td>4</td>
<td>1%</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td>528</td>
<td>100%</td>
</tr>
</tbody>
</table>

### 28. The telephone system in my residence is sufficient for my needs.

<table>
<thead>
<tr>
<th>Rating</th>
<th>Count</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Strongly Disagree</td>
<td>35</td>
<td>7%</td>
</tr>
<tr>
<td>Disagree</td>
<td>48</td>
<td>9%</td>
</tr>
<tr>
<td>Neutral</td>
<td>114</td>
<td>22%</td>
</tr>
<tr>
<td>Agree</td>
<td>137</td>
<td>26%</td>
</tr>
<tr>
<td>Strongly Agree</td>
<td>41</td>
<td>8%</td>
</tr>
<tr>
<td>* N/A *</td>
<td>152</td>
<td>29%</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td>527</td>
<td>100%</td>
</tr>
</tbody>
</table>
29. Please select how you make outside calls from within the IIT campus.

<table>
<thead>
<tr>
<th>Method</th>
<th>Count</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Calling Card</td>
<td>78</td>
<td>7%</td>
</tr>
<tr>
<td>Cellular Phone</td>
<td>1004</td>
<td>89%</td>
</tr>
<tr>
<td>Computer Software</td>
<td>113</td>
<td>10%</td>
</tr>
<tr>
<td>Do not make outside calls from IIT</td>
<td>79</td>
<td>7%</td>
</tr>
<tr>
<td>Other, Please Specify</td>
<td>9</td>
<td>1%</td>
</tr>
</tbody>
</table>

30. Please select how you make on-campus calls from within the IIT campus?

<table>
<thead>
<tr>
<th>Method</th>
<th>Count</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Calling Card</td>
<td>10</td>
<td>1%</td>
</tr>
<tr>
<td>Cellular phone</td>
<td>842</td>
<td>75%</td>
</tr>
<tr>
<td>Computer software</td>
<td>24</td>
<td>2%</td>
</tr>
<tr>
<td>Hall phone</td>
<td>78</td>
<td>7%</td>
</tr>
<tr>
<td>Do not make internal IIT calls</td>
<td>219</td>
<td>19%</td>
</tr>
<tr>
<td>Other, please specify</td>
<td>45</td>
<td>4%</td>
</tr>
</tbody>
</table>

31. I would be willing to change my cell phone provider for a better deal (while keeping the same number).

<table>
<thead>
<tr>
<th>Response</th>
<th>Count</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Strongly Disagree</td>
<td>164</td>
<td>15%</td>
</tr>
<tr>
<td>Disagree</td>
<td>180</td>
<td>16%</td>
</tr>
<tr>
<td>Neutral</td>
<td>241</td>
<td>21%</td>
</tr>
<tr>
<td>Agree</td>
<td>265</td>
<td>23%</td>
</tr>
<tr>
<td>Strongly Agree</td>
<td>257</td>
<td>23%</td>
</tr>
<tr>
<td>Do not use a cell phone</td>
<td>23</td>
<td>2%</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td>1130</td>
<td>100%</td>
</tr>
</tbody>
</table>

32. Have any of the courses you've taken at IIT had an Internet section?

<table>
<thead>
<tr>
<th>Response</th>
<th>Count</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yes</td>
<td>519</td>
<td>46%</td>
</tr>
<tr>
<td>No</td>
<td>510</td>
<td>45%</td>
</tr>
<tr>
<td>Don't know</td>
<td>106</td>
<td>9%</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td>1135</td>
<td>100%</td>
</tr>
</tbody>
</table>
33. If you have taken an internet course section, did you:

<table>
<thead>
<tr>
<th>Option</th>
<th>Count</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Attend the live class and use the Internet video of the</td>
<td>300</td>
<td>58%</td>
</tr>
<tr>
<td>lectures for review</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Attend the live class and never use the Internet video of</td>
<td>103</td>
<td>20%</td>
</tr>
<tr>
<td>the lectures</td>
<td></td>
<td></td>
</tr>
<tr>
<td>View the class only via the Internet</td>
<td>166</td>
<td>32%</td>
</tr>
</tbody>
</table>

34. If IIT offered an internet section for your courses, would you:

<table>
<thead>
<tr>
<th>Option</th>
<th>Count</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Attend the live class and use the Internet video of the</td>
<td>848</td>
<td>76%</td>
</tr>
<tr>
<td>lectures for review</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Attend the live class and never use the Internet video of</td>
<td>143</td>
<td>13%</td>
</tr>
<tr>
<td>the lectures for review</td>
<td></td>
<td></td>
</tr>
<tr>
<td>View the class only over the internet</td>
<td>211</td>
<td>19%</td>
</tr>
</tbody>
</table>

35. IIT classrooms have sufficient audio/visual equipment to meet my academic needs.

<table>
<thead>
<tr>
<th>Option</th>
<th>Count</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Strongly Disagree</td>
<td>29</td>
<td>3%</td>
</tr>
<tr>
<td>Disagree</td>
<td>110</td>
<td>10%</td>
</tr>
<tr>
<td>Neutral</td>
<td>248</td>
<td>22%</td>
</tr>
<tr>
<td>Agree</td>
<td>576</td>
<td>51%</td>
</tr>
<tr>
<td>Strongly Agree</td>
<td>142</td>
<td>13%</td>
</tr>
<tr>
<td><em>N/A</em></td>
<td>23</td>
<td>2%</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td>1128</td>
<td>100%</td>
</tr>
</tbody>
</table>

36. I would like to use a handheld device to view or listen to podcasts of class lectures, special IIT events and news items.

<table>
<thead>
<tr>
<th>Option</th>
<th>Count</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Strongly Disagree</td>
<td>47</td>
<td>4%</td>
</tr>
<tr>
<td>Disagree</td>
<td>123</td>
<td>11%</td>
</tr>
<tr>
<td>Neutral</td>
<td>302</td>
<td>27%</td>
</tr>
<tr>
<td>Agree</td>
<td>400</td>
<td>36%</td>
</tr>
<tr>
<td>Strongly Agree</td>
<td>216</td>
<td>19%</td>
</tr>
</tbody>
</table>
37. Do you own any of the following (Select all that apply):

<table>
<thead>
<tr>
<th>Device Type</th>
<th>Count</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Desktop Computer</td>
<td>482</td>
<td>43%</td>
</tr>
<tr>
<td>Laptop Computer</td>
<td>1006</td>
<td>90%</td>
</tr>
<tr>
<td>Video iPod or similar mobile device</td>
<td>439</td>
<td>39%</td>
</tr>
<tr>
<td>Audio-only iPod or other mp3 player</td>
<td>494</td>
<td>44%</td>
</tr>
</tbody>
</table>

38. How long have you been studying at IIT?

<table>
<thead>
<tr>
<th>Duration</th>
<th>Count</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>0-1 years</td>
<td>466</td>
<td>41%</td>
</tr>
<tr>
<td>1-2 years</td>
<td>304</td>
<td>27%</td>
</tr>
<tr>
<td>2-3 years</td>
<td>158</td>
<td>14%</td>
</tr>
<tr>
<td>3-4 years</td>
<td>115</td>
<td>10%</td>
</tr>
<tr>
<td>4-5 years</td>
<td>54</td>
<td>5%</td>
</tr>
<tr>
<td>5+ years</td>
<td>32</td>
<td>3%</td>
</tr>
</tbody>
</table>

Total: 1129 responses, 100%

39. Please enter your e-mail address if you would like to be considered for the prize drawing:

1071 responses

40. Please feel free to provide any additional comments or technology-related concerns you may have in the box below:

326 responses
## 2006-2008 Building Technology
### Infrastructure Upgrades

<table>
<thead>
<tr>
<th>Updates:</th>
<th>E1</th>
<th>Stuart</th>
<th>Metals</th>
<th>Siegel Hall</th>
<th>Life Sciences</th>
<th>Perlstein</th>
<th>Crown Hall</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fiber connection into the buildings</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>Fiber raiser between floors</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>New teledata closets</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>New network switches</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>Infrastructure for Distance Learning</td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>A/V equipped classrooms/labs</td>
<td>Y</td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Full wireless coverage</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>Replacement of CAT3 cables with CAT6 cables</td>
<td>Partial</td>
<td>Partial</td>
<td>Partial</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>Partial</td>
</tr>
<tr>
<td>UPS in teledata</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>Building</td>
<td>Room Number</td>
<td>In-Service Date</td>
<td>Type</td>
<td>Basic A/V</td>
<td>Distance Learning</td>
<td>Projector Model #</td>
<td>Lens Size - Proj to Screen</td>
</tr>
<tr>
<td>----------</td>
<td>-------------</td>
<td>----------------</td>
<td>-----------------</td>
<td>-----------</td>
<td>------------------</td>
<td>-------------------</td>
<td>--------------------------</td>
</tr>
<tr>
<td>E1</td>
<td>25</td>
<td></td>
<td>classroom</td>
<td>x</td>
<td>x</td>
<td>christie LX40</td>
<td>short - 10</td>
</tr>
<tr>
<td></td>
<td>29</td>
<td></td>
<td>lab</td>
<td>x</td>
<td>x</td>
<td>christie LX40</td>
<td>short - 20</td>
</tr>
<tr>
<td></td>
<td>34</td>
<td>2006</td>
<td>classroom</td>
<td>x</td>
<td>x</td>
<td>christie LX40</td>
<td>short - 10</td>
</tr>
<tr>
<td></td>
<td>102</td>
<td></td>
<td>classroom</td>
<td>x</td>
<td>x</td>
<td>christie LX40</td>
<td>short - 10</td>
</tr>
<tr>
<td></td>
<td>103</td>
<td></td>
<td>classroom</td>
<td>x</td>
<td>x</td>
<td>christie LX40</td>
<td>short - 10</td>
</tr>
<tr>
<td></td>
<td>104</td>
<td></td>
<td>classroom</td>
<td>x</td>
<td>x</td>
<td>christie LX40</td>
<td>short - 10</td>
</tr>
<tr>
<td></td>
<td>106</td>
<td></td>
<td>classroom</td>
<td>x</td>
<td>x</td>
<td>christie LX40</td>
<td>short - 10</td>
</tr>
<tr>
<td></td>
<td>119</td>
<td></td>
<td>classroom</td>
<td>x</td>
<td>x</td>
<td>christie LX40</td>
<td>short - 10</td>
</tr>
<tr>
<td></td>
<td>121</td>
<td></td>
<td>classroom</td>
<td>x</td>
<td>x</td>
<td>christie LX40</td>
<td>short - 10</td>
</tr>
<tr>
<td></td>
<td>122</td>
<td></td>
<td>classroom</td>
<td>x</td>
<td>x</td>
<td>christie LX40</td>
<td>short - 10</td>
</tr>
<tr>
<td></td>
<td>123</td>
<td></td>
<td>classroom</td>
<td>x</td>
<td>x</td>
<td>christie LX40</td>
<td>short - 15</td>
</tr>
<tr>
<td></td>
<td>124</td>
<td></td>
<td>classroom</td>
<td>x</td>
<td>x</td>
<td>christie LX40</td>
<td>short - 10</td>
</tr>
<tr>
<td></td>
<td>141</td>
<td></td>
<td>specialty room</td>
<td>x</td>
<td>x</td>
<td>christie LX40</td>
<td>short - 10</td>
</tr>
<tr>
<td></td>
<td>241</td>
<td></td>
<td>classroom</td>
<td>x</td>
<td>x</td>
<td>christie LX40</td>
<td>short - 10</td>
</tr>
<tr>
<td></td>
<td>242</td>
<td></td>
<td>classroom</td>
<td>x</td>
<td>x</td>
<td>christie LX40</td>
<td>short - 10</td>
</tr>
<tr>
<td></td>
<td>244</td>
<td></td>
<td>classroom</td>
<td>x</td>
<td>x</td>
<td>christie LX40</td>
<td>short - 10</td>
</tr>
<tr>
<td></td>
<td>245</td>
<td></td>
<td>classroom</td>
<td>x</td>
<td>x</td>
<td>christie LX40</td>
<td>short - 10</td>
</tr>
<tr>
<td></td>
<td>104</td>
<td>2007</td>
<td>auditorium</td>
<td>x</td>
<td>x</td>
<td>Sanyo PLC-XT25L</td>
<td>short-10</td>
</tr>
<tr>
<td>Stuart</td>
<td>108</td>
<td>2006</td>
<td>lab</td>
<td>x</td>
<td>x</td>
<td>christie LX 40</td>
<td>short-10</td>
</tr>
<tr>
<td></td>
<td>111 SB</td>
<td>2006</td>
<td>classroom</td>
<td>x</td>
<td>x</td>
<td>Sanyo PLC-XT25L</td>
<td>short-10</td>
</tr>
<tr>
<td></td>
<td>112 E</td>
<td>2007</td>
<td>lab</td>
<td>x</td>
<td>x</td>
<td>Sanyo PLC-XT25L</td>
<td>short-10</td>
</tr>
<tr>
<td></td>
<td>112 F</td>
<td>2007</td>
<td>lab</td>
<td>x</td>
<td>x</td>
<td>Sanyo PLC-XT25L</td>
<td>short-10</td>
</tr>
<tr>
<td></td>
<td>112 J</td>
<td>2006</td>
<td>lab</td>
<td>x</td>
<td>x</td>
<td>Sanyo PLC-XT25L</td>
<td>short-10</td>
</tr>
<tr>
<td></td>
<td>113</td>
<td>2006</td>
<td>classroom</td>
<td>x</td>
<td>x</td>
<td>Sanyo PLC-XT25L</td>
<td>short-10</td>
</tr>
<tr>
<td></td>
<td>201</td>
<td>2007</td>
<td>classroom</td>
<td>x</td>
<td>x</td>
<td>Sanyo PLC-XT25L</td>
<td>short-10</td>
</tr>
<tr>
<td></td>
<td>204</td>
<td>2007</td>
<td>classroom</td>
<td>x</td>
<td>x</td>
<td>Sanyo PLC-XT25L</td>
<td>short-10</td>
</tr>
<tr>
<td></td>
<td>212</td>
<td>2007</td>
<td>classroom</td>
<td>x</td>
<td>x</td>
<td>Sanyo PLC-XT25L</td>
<td>short-10</td>
</tr>
<tr>
<td></td>
<td>213</td>
<td>2007</td>
<td>classroom</td>
<td>x</td>
<td>x</td>
<td>Sanyo PLC-XT25L</td>
<td>short-10</td>
</tr>
<tr>
<td></td>
<td>220</td>
<td>2007</td>
<td>classroom</td>
<td>x</td>
<td>x</td>
<td>Sanyo PLC-XT25L</td>
<td>short-10</td>
</tr>
<tr>
<td></td>
<td>223</td>
<td>2007</td>
<td>classroom</td>
<td>x</td>
<td>x</td>
<td>Sanyo PLC-XT25L</td>
<td>short-10</td>
</tr>
<tr>
<td></td>
<td>225</td>
<td>2007</td>
<td>classroom</td>
<td>x</td>
<td>x</td>
<td>Sanyo PLC-XT25L</td>
<td>short-10</td>
</tr>
<tr>
<td>LS</td>
<td>121</td>
<td>OLD classroom</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>none - TV monitors</td>
<td></td>
</tr>
<tr>
<td>PH</td>
<td>131</td>
<td>OLD auditorium</td>
<td>x</td>
<td>Sony XGA VPL-4X51</td>
<td>short - 20</td>
<td>LMP-F300</td>
<td>EV4000AF</td>
</tr>
<tr>
<td>DVD Model #</td>
<td>Audio Amplifier</td>
<td>Media Processor</td>
<td>Crestron Control Panel System</td>
<td>Surge Eliminator</td>
<td>VDA (Video Distribution Amplifier)</td>
<td>Screen Width</td>
<td>Dist. From Floor</td>
</tr>
<tr>
<td>-------------</td>
<td>-----------------</td>
<td>-----------------</td>
<td>------------------------------</td>
<td>-----------------</td>
<td>-----------------------------------</td>
<td>--------------</td>
<td>-----------------</td>
</tr>
<tr>
<td>AG-VP320</td>
<td>Crown 180A</td>
<td>MP2E</td>
<td>RS-232/422 COM ST-CS</td>
<td>SX-1115-RT</td>
<td>VDA-16 (2)</td>
<td>8</td>
<td>7</td>
</tr>
<tr>
<td>AG-VP320</td>
<td>Crown 180A</td>
<td>MP2E</td>
<td>RS-232/422 COM ST-CS</td>
<td>SX-1115-RT</td>
<td>VDA-16 (2)</td>
<td>8</td>
<td>7</td>
</tr>
<tr>
<td>AG-VP320</td>
<td>Crown 180A</td>
<td>MP2E</td>
<td>RS-232/422 COM ST-CS</td>
<td>SX-1115-RT</td>
<td>VDA-16 (2)</td>
<td>8</td>
<td>7</td>
</tr>
<tr>
<td>AG-VP320</td>
<td>Crown 180A</td>
<td>MP2E</td>
<td>RS-232/422 COM ST-CS</td>
<td>SX-1115-RT</td>
<td>VDA-16 (2)</td>
<td>8</td>
<td>7</td>
</tr>
<tr>
<td>AG-VP320</td>
<td>Crown 180A</td>
<td>MP2E</td>
<td>RS-232/422 COM ST-CS</td>
<td>SX-1115-RT</td>
<td>VDA-16 (2)</td>
<td>8</td>
<td>7</td>
</tr>
<tr>
<td>AG-VP320</td>
<td>Crown 180A</td>
<td>MP2E</td>
<td>RS-232/422 COM ST-CS</td>
<td>SX-1115-RT</td>
<td>VDA-16 (2)</td>
<td>8</td>
<td>7</td>
</tr>
<tr>
<td>AG-VP320</td>
<td>Crown 180A</td>
<td>MP2E</td>
<td>RS-232/422 COM ST-CS</td>
<td>SX-1115-RT</td>
<td>VDA-16 (2)</td>
<td>8</td>
<td>7</td>
</tr>
<tr>
<td>AG-VP320</td>
<td>Crown 180A</td>
<td>MP2E</td>
<td>RS-232/422 COM ST-CS</td>
<td>SX-1115-RT</td>
<td>VDA-16 (2)</td>
<td>8</td>
<td>6.5</td>
</tr>
<tr>
<td>AG-VP320</td>
<td>Crown 180A</td>
<td>MP2E</td>
<td>RS-232/422 COM ST-CS</td>
<td>SX-1115-RT</td>
<td>VDA-16 (2)</td>
<td>8</td>
<td>6.5</td>
</tr>
<tr>
<td>D380P</td>
<td>Extrtron RGB CP2E</td>
<td>SX-1115-RT (2)</td>
<td>SX-1115-RT (2)</td>
<td>SX-1115-RT (2)</td>
<td>SX-1115-RT (2)</td>
<td>10 or 12</td>
<td>12</td>
</tr>
<tr>
<td>D380P</td>
<td>Extrtron RGB CP2E</td>
<td>SX-1115-RT (2)</td>
<td>SX-1115-RT (2)</td>
<td>SX-1115-RT (2)</td>
<td>SX-1115-RT (2)</td>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td>AG-VP320</td>
<td>Crown 180A</td>
<td>MP2E</td>
<td>RS-232/422 COM ST-CS</td>
<td>SX-1115-RT</td>
<td>VDA-16 (2)</td>
<td>8</td>
<td>10</td>
</tr>
<tr>
<td>AG-VP320</td>
<td>Crown 180A</td>
<td>MP2E</td>
<td>RS-232/422 COM ST-CS</td>
<td>SX-1115-RT</td>
<td>VDA-16 (2)</td>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td>AG-VP320</td>
<td>Crown 180A</td>
<td>MP2E</td>
<td>RS-232/422 COM ST-CS</td>
<td>SX-1115-RT</td>
<td>VDA-16 (2)</td>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td>AG-VP320</td>
<td>Crown 180A</td>
<td>MP2E</td>
<td>RS-232/422 COM ST-CS</td>
<td>SX-1115-RT</td>
<td>VDA-16 (2)</td>
<td>8</td>
<td>9</td>
</tr>
<tr>
<td>AG-VP320</td>
<td>Crown 180A</td>
<td>MP2E</td>
<td>RS-232/422 COM ST-CS</td>
<td>SX-1115-RT</td>
<td>VDA-16 (2)</td>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td>AG-VP320</td>
<td>Crown 180A</td>
<td>MP2E</td>
<td>RS-232/422 COM ST-CS</td>
<td>SX-1115-RT</td>
<td>VDA-16 (2)</td>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td>AG-VP320</td>
<td>Crown 180A</td>
<td>MP2E</td>
<td>RS-232/422 COM ST-CS</td>
<td>SX-1115-RT</td>
<td>VDA-16 (2)</td>
<td>8</td>
<td>6</td>
</tr>
<tr>
<td>AG-VP320</td>
<td>Crown 180A</td>
<td>MP2E</td>
<td>RS-232/422 COM ST-CS</td>
<td>SX-1115-RT</td>
<td>VDA-16 (2)</td>
<td>8</td>
<td>6</td>
</tr>
<tr>
<td>AG-VP320</td>
<td>Crown 180A</td>
<td>MP2E</td>
<td>RS-232/422 COM ST-CS</td>
<td>SX-1115-RT</td>
<td>VDA-16 (2)</td>
<td>8</td>
<td>6</td>
</tr>
<tr>
<td>AG-VP320</td>
<td>Crown 180A</td>
<td>MP2E</td>
<td>RS-232/422 COM ST-CS</td>
<td>SX-1115-RT</td>
<td>VDA-16 (2)</td>
<td>8</td>
<td>6</td>
</tr>
<tr>
<td>AG-VP320</td>
<td>Crown 180A</td>
<td>MP2E</td>
<td>RS-232/422 COM ST-CS</td>
<td>SX-1115-RT</td>
<td>VDA-16 (2)</td>
<td>8</td>
<td>6</td>
</tr>
<tr>
<td>AG-VP320</td>
<td>Crown 180A</td>
<td>MP2E</td>
<td>RS-232/422 COM ST-CS</td>
<td>SX-1115-RT</td>
<td>VDA-16 (2)</td>
<td>8</td>
<td>6</td>
</tr>
<tr>
<td>AG-VP320</td>
<td>Crown 180A</td>
<td>MP2E</td>
<td>RS-232/422 COM ST-CS</td>
<td>SX-1115-RT</td>
<td>VDA-16 (2)</td>
<td>8</td>
<td>6</td>
</tr>
<tr>
<td>AG-VP320</td>
<td>Crown 180A</td>
<td>MP2E</td>
<td>RS-232/422 COM ST-CS</td>
<td>SX-1115-RT</td>
<td>VDA-16 (2)</td>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td>AG-VP320</td>
<td>Crown 180A</td>
<td>MP2E</td>
<td>RS-232/422 COM ST-CS</td>
<td>SX-1115-RT</td>
<td>VDA-16 (2)</td>
<td>8</td>
<td>6</td>
</tr>
</tbody>
</table>

Appendix IV Facilities

<table>
<thead>
<tr>
<th>DVD Model #</th>
<th>Audio Amplifier</th>
<th>Media Processor</th>
<th>Crestron Control Panel System</th>
<th>Surge Eliminator</th>
<th>VDA (Video Distribution Amplifier)</th>
<th>Screen Width</th>
<th>Dist. From Floor</th>
<th>Dist. from Wall</th>
</tr>
</thead>
<tbody>
<tr>
<td>Software</td>
<td>Installed Version</td>
<td>SH237</td>
<td>SH237</td>
<td>HUB112 RICE</td>
<td>3410</td>
<td>CR501</td>
<td>SH235</td>
<td>E1029</td>
</tr>
<tr>
<td>--------------------------</td>
<td>-------------------</td>
<td>-------</td>
<td>-------</td>
<td>------------</td>
<td>------</td>
<td>-------</td>
<td>-------</td>
<td>-------</td>
</tr>
<tr>
<td>Adobe Illustrator CS 3</td>
<td></td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Adobe InDesign CS 5</td>
<td></td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Adobe Premiere CS 5</td>
<td></td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Adobe Reader CS 5</td>
<td></td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ap2 2007</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>AutoCAD Electrical 2008</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X X X X X X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AutoCAD Map 3D 2008</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X X X X X X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AutoCAD Mechanical 2008</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X X X X X X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AutoCAD Professional 2006</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X X X X X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AutoCAD Revit Design 2006</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X X X X X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Autodesk Inventor 2007</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X X X X X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Autodesk VIZ 2008</td>
<td></td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Basic Stamp Editor 2.2.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>Bentley View XM 2.89.03.06</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Blocks</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>Blender 2.41</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>Cat-A-Soft 8.6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>CeltixTools 1.18</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>CINEMA 4D</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>COM/Check 1.1.v</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>Corel Graphics Suite 1.1</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X X X X X X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CorelDRAW 11</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>CorelDRAW Suite 2009</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>DraftSight</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>Eagle 4.20</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>eBuilder/SoftwareCAD III</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X X X X X X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FileMaker</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>FramePlus 3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>GOOGLE EARTH</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>Gravity Sketch 2.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>GrowthLab 3.3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>HyperWorks 2006</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>InConTech 3.03</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>iTextBrowser/SoftwareCAD</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>InverseWorks 4.1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>Jira</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>K2I</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>K2I</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>Linux/VIEW 2.1.0</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X X X X X X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Microsoft Office Professional Edition</td>
<td>2003 SP3</td>
<td>X</td>
<td>X</td>
<td>X X X X X X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Microsoft Visual Studio 2005 SP3</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>ModelSim SE PLUS 1.0a</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X X X X X X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MSProject 2000</td>
<td></td>
<td>1.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>NotesLAB 13.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>Office 11.0.0 Demo</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X X X X X X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Panorama</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>Pro Engineer Wildfire 2</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X X X X X X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pro Mechanica Wildfire 2</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X X X X X X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Print Express 1.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>Print Programmer 2.0.0.4.1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>Print Designer 4.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>Print Designer EP CSD</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>Quantum GIS 3.5.1</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X X X X X X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Revit Architecture 2008</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X X X X X X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ricoh Aficio X X X</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X X X X X X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SAF2000 11</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>SAF3000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>SAW-Path/ViewRay 3.1</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X X X X X X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Seewe 2.2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>Silvers 2.14</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>TCA Select</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>VisualDOC 2.2</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X X X X X X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>WINDOW 3.1</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X X X X X X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>WUTI 4.5</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X X X X X X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Alice</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X X X X X X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Key: 112EFX, 112H, 112X, 112Y, 112Z

Appendix -IV Facilities

Addendum E, page 29
<table>
<thead>
<tr>
<th>Software</th>
<th>Installed Version</th>
<th>Dept</th>
<th>Key: 112E, 112F, 112X</th>
<th>SH237</th>
<th>112J</th>
<th>HUB112/RICE</th>
<th>Sh410</th>
<th>CRC01</th>
<th>SH236</th>
<th>E1029</th>
<th>MSV</th>
<th>AM218</th>
<th>MTCC Cafeteria</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 7-Zip</td>
<td>4.42</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2 Adobe Creative Suite</td>
<td>CS 3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3 Adobe Pagemaker</td>
<td>7.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4 Adobe Premiere</td>
<td>8.1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5 Adobe Reader</td>
<td>8.1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6 Apache Tomcat</td>
<td>6.0.13</td>
<td>CS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7 Citrix Presentation Server</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8 Core FTP Pro</td>
<td>1.3c.build 1447.6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9 Coral Graphics Suite</td>
<td>11</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10 Cyberlink Power DVD X</td>
<td>9.5.2806</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11 Eclipse</td>
<td>3.3.0</td>
<td>CS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12 EditPlus</td>
<td>2.31</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>13 Google Earth</td>
<td>4.1.7087.5048 (beta)</td>
<td>CAEE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14 HAM</td>
<td>2.8</td>
<td>CS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15 iTunes</td>
<td>7.2.0.54</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>16 J2EE</td>
<td>Java Application Platform SDK U2</td>
<td>CS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17 JacORB</td>
<td>1.4.1</td>
<td>CS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>18 JDK</td>
<td>1.6.0.01</td>
<td>CS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>19 LEEdit</td>
<td>4.2</td>
<td>ITM</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20 JMP</td>
<td>7</td>
<td>MATH</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>21 Jython</td>
<td>2.2 RC3</td>
<td>CS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>22 Mathcad</td>
<td>11.02</td>
<td>MATH</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>23 Mathlab</td>
<td>7.4 R2007a</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>24 MAX+plus II</td>
<td>10.2</td>
<td>CS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>25 McAfee VirusScan</td>
<td>8.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>26 Microsoft Frontpage</td>
<td>2000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>27 Microsoft Office Professional Edition</td>
<td>2003 SP2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>28 Microsoft Project Professional</td>
<td>2003 SP3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>29 Microsoft Visio Professional</td>
<td>2003 SP3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>30 Microsoft Visual Studio</td>
<td>2005 SP1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>31 Microsoft Windows XP Professional</td>
<td>SP2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>32 Mozilla Firefox</td>
<td>2.0.0.4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>33 Mozilla Thunderbird</td>
<td>2005</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>34 Mozilla Thunderbird</td>
<td>2005</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>35 Oracle Database 10g Client</td>
<td>10.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>36 Oracle SQL Developer</td>
<td>10.0.15.57</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>37 Paint Shop Pro Photo</td>
<td>9.0</td>
<td>ITM</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>38 PCSpim</td>
<td>7.2</td>
<td>CS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>39 PLT Scheme</td>
<td>370</td>
<td>CS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>40 PowerTerm Pro Enterprise</td>
<td>8.3.3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>41 Python</td>
<td>2.5.1</td>
<td>CS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>42 QuickTime</td>
<td>7.1.6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>43 Real Player</td>
<td>10.5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>44 Roxio Dell Edition</td>
<td>2.4.32A</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>45 SecureCRT</td>
<td>3.2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>46 SPSS 15.0 for Windows</td>
<td>15.0.1.1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>47 Subversion</td>
<td>1.4.4</td>
<td>CS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>48 WinSCP</td>
<td>3.8.2</td>
<td>ITM</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>49 X-Win32</td>
<td>5.1.2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Key: 112E, 112F, 112X
SH237 Siegel Hall
112J Stuart Building
HUB112/RICE Hermann Union Building/ Rice Campus
Sh410 410 State Street
CRC01 Crown Hall
SH236 Siegel Hall
E1029 Engineering 1
MSV McCormick Student Village
AM218 Alumni Memorial
MTCC Cafeteria McCormick Tribune Campus Center
Appendix - IV Facilities
1. Wear solid colors, no white, no black, and no patterns/stripes/plaids.

2. Start and End on Time - You should end your class with a closing statement, on time. Class broadcasts will be terminated no more than 4 minutes after your scheduled class end time – even if it is the last scheduled for the day. After a 2-minute grace period, the TD (technical director) will start to close down the class video and audio and by the 4th overtime minute, broadcast will end. Remote students will not see or hear anything after that point. This cut-off is necessary for staff and technical reasons.

3. Test your PC in your classroom before using it in class so you know the toggle switch to use for display to both the PC monitor and to send VGA output to the projector. Blue or gray backgrounds work best; white backgrounds are too hot and may not be broadcast. Use a sans serif font such as Arial, 18-24 points minimum, no shadows, 5-6 bullets per page – do not use 20% outer perimeter due to different resolutions between PC and TVs. IIT Online provides a sample template and instructions at www.iit-online.iit.edu/faculty.

4. Keep the microphone pointing up and about 6” beneath your chin, centered.

5. Repeat student questions. If there is a phone call, you may need to ask for the volume to be adjusted.

6. Print one copy of your class materials for display on the document camera using Arial, 18 points or more, with 2” margins left/right, on pastel-colored paper. Blue is best as there is no bleed-in from the sheets behind. Interleave blank pages if using other colors. To put in focus on the document camera, zoom in all the way, press Auto-focus once, and when the camera completes the focus, zoom out. If the depth of materials changes to a large extent, refocus.

7. When using the whiteboards, keep within the tape or divider boundaries on any one board. This is all that can be on camera at one time.

8. Check the instructor monitor often to be sure papers are aligned horizontally, focus is set and zoom is appropriate. If you are acting as your own technical director, return to your image often. Students want to see you, and benefit from your enthusiasm.

9. If your class is also on the Internet, place all written materials, and printouts from the electronic whiteboard in the envelope provided. Internet development staff will return all papers to you.

10. Please let the TD know of any special requests. He/she is also an IIT student, and many are new this semester. While each TD has completed training, they will need your patience while they learn your individual teaching style.

11. Use the yellow faculty response card for any comments or suggestions. The TD will return it to the Chief Engineer, or you may send it directly.

12. Above all, be calm. Please remember you are on-air and students are looking to you for the proper response to the unexpected. You can always take a break, and when off camera, deal with any situation. If there is a harassing call received, ask for it to be dropped. If this occurs again, ask remote students to wait until after class to call and ask the TD to stop incoming calls. We are here to help. There is ALWAYS staff present to assist. Thank you!
IIT Online Technical Services and Support Services

The following procedures have been instituted in order to meet the established university policies and deadlines, and to meet the needs and expectations of students and faculty. They are divided into procedures for televised and videoconferencing courses, and Internet courses.

I. Procedures for Televised and Videoconferencing Courses

1. **IIT Online Responsibilities** – When you arrive at your broadcast/videoconferencing room you should expect to find all technical equipment turned-on, the document camera set for auto-focus, a legal pad and pen, markers and eraser for the board, microphone, and remote connections active. A class identification screen (from a character generator or on the document camera) should have your class correctly designated.

Orientation sessions and/or one-on-one scheduled sessions on the use of the equipment, and techniques for teaching on camera, are provided each semester, and are also available on the web at [www.iit-online.iit.edu/faculty](http://www.iit-online.iit.edu/faculty).

2. **Timeliness** - It is extremely important that broadcasts begin and end on time. The broadcast schedule of courses allows for only a ten-minute break between classes. Timing of classes being recorded is automatic, and cannot begin prior to the designated time, nor extend beyond the end time. During the ten minutes planned between classes, boards are cleaned, pens tested, batteries replaced, sound checks conducted, video and audio recordings saved, and a host of additional technical tasks. Your students are expecting classes to start and end on time. Please be ready to set up for your class no later than 4 minutes prior to your start time, and end on time. If you go over by even one minute, it cuts down the time we have to set up for the next class, limits the set-up time of the next faculty member in that room, or delays the mandatory end-of-day broadcast activities that must be performed on schedule.

   **Video and audio will be cut off the air at 4 minutes over the scheduled end-of-class time.**

3. **Special Requests** - If you have special requests such as an advance taping or videotape to be played in class, or are using a laptop for the first time in a particular broadcast room, let us know before your class so we have time to assist before you are on the air. We will try and accommodate any special requests received at IIT Online 24-48 hours prior to your class session.

4. **Faculty Responsibilities** – The cameraperson, or TD (technical director) assigned to your broadcast or taping is usually an IIT student. This may be their first on-campus job, and they may even be new to the United States. They each receive training before the semester begins, and their supervisors are always available during the broadcast. Please be patient with them. If you will be using one technology such as the white boards only, or your laptop, or a mix, let them know. They will then be more prepared to meet the camera needs for your specific class.

If you have concerns or suggestions that are not immediate, use the yellow Faculty Response Cards placed in the room for every broadcast. The staff is always available for assistance by phone or email. These are returned to the Chief Engineer and he or the Supervisor will respond to your comments. If you have an immediate need such as additional markers, just direct them to the TD verbally and he or she will respond.
Attached is a one-page “Best Practices on Camera” document. It contains suggestions for video, audio, and class materials so that they are broadcast or posted on the Internet with the highest quality possible. Please pay particular attention to the placement and use of the microphone, writing on the board within one panel at a time, and the absence of white, black, and patterned/striped clothing on camera. Please take the few minutes needed to read this important document.

II. Procedures for Internet Courses

The Internet Development staff will contact every faculty member whose class is on the Internet. Our intent is to work with you in the design of your course on the Internet.

1. IIT Online Technical Services Posting Responsibilities – Course lectures and the .pdf of the slides (if applicable) will be available on the Blackboard web page through the IIT-Online tab 24-48 hours after the live class session. Each class will be available 24/7 for the duration of the semester. Video/audio and materials in this section will be deleted the Friday following the last university final exam date.

2. Faculty Responsibilities for Internet Courses

a. Mode of Delivery – The standard methodology is streamed video and audio with synchronized slides (samples are at www.iit-online.iit.edu, select Demo.) The slide default is created from “screen grabs” of the instructor’s writing on the whiteboard or scans of printed material from the class, from instructor’s own PC content etc. Faculty teaching on the Internet who wish assistance in developing materials should contact Louise Hewitt (x75167 hewitt@iit.edu). Besides these two modes, there are other options for web-enabling course, such as faculty providing all materials to the students, and video-only streaming of the lectures. Faculty may use any of these services, but should work with Billy to discuss the best option for their class in advance of the beginning of the semester.

b. Posted Materials – In order for IIT Online to produce the best quality product for the Internet, faculty, their selected guest speakers, and students making presentations in-class should provide all class materials to IIT Online Technical Services (103 Stuart Building, fax to x75913, or email to onlinematerials@iit.edu) no later than 1:00pm on the following day. These materials include handouts, hand-written notes, web pages, electronic materials, and printouts from the electronic whiteboards. (Use the electronic whiteboards and print the board particularly when drawing tables, charts, or other complex images.) For convenience, these may be inserted into the envelope provided in the broadcast room or they may be brought to IIT Online (103 Stuart) before or immediately after each class or emailed. For Rice and Moffett campuses, hard-copy materials should be provided to the technical director who will fax them to main campus that day. This will allow for scanning, editing for brightness and synchronization of slides with the video and audio, and posting within the maximum of 48 hours after the live class. All items sent to IIT Online will be returned.

c. Handouts and other materials may be posted by the faculty on the course Blackboard page.

d. Distribution of Materials – Handouts for distribution to remote TV sites and other locations should be received by IIT Online Support Services (103 Stuart) preferably two working days in advance to insure that the students receive the material prior to the class. Duplication and assistance in preparing inter-campus, and remote site delivery envelopes
are available by contacting Chuck Scott (x75217 or scott@iit.edu) or Gwen Bowling (x77957 or bowling@iit.edu) located in 103SB.

e. **Exams** - Faculty members need to provide exam dates to Chuck Scott at IIT Online Support Services (x75217 or scott@iit.edu) two weeks prior to the exam date and deliver the exam to Chuck two working days before the exam date. One exception is that Monday exams must be delivered to the IIT Online Support Services Office at 103 Stuart on Wednesday, because the couriers do not deliver to all sites on Fridays. In addition, it is important that instructors teaching courses that meet two days a week schedule all exams on the first meeting day of the week so that Internet students have sufficient time (usually the weekend) to review the lecture material. For example if the class is taught on Tuesday and Thursday, the exam should be on Tuesday.

   **IIT Online is unable to honor instructor requests to post a class earlier than 48 hours so that an exam schedule can be met. It is the responsibility of the instructor to schedule exams as explained above.**

3. **Student Accessibility and Responsibilities** –

   a. **Access** - All students (live, TV remote, and Internet) enrolled in a class supported on the Internet who appear on the official roster for the class will have their account to view the lectures enabled. Students are directed to a document explaining their responsibilities for participating in the IIT Online Internet system, keeping up with viewing the lectures and the course requirements that may be in your syllabus. Links are provided on the Blackboard, IIT-Online tab. The Internet material includes video and/or slides, if appropriate. Instructors may elect to provide links to their own homepages or use other posting mechanisms independently of IIT Online.

   b. **Exams** - Students taking courses via the Internet are required to take exams on the same day as the students in the live and TV sections of that same course. Instructors may make exceptions, as they would in the live or TV course to accommodate a student’s needs, but all students will be expected to take exams on the same day. Students who are not near an IIT site must contact a reputable testing center (usually a community college) in their area and then contact Chuck Scott at (x75217 or scott@iit.edu) IIT Online Support Services to finalize the details of the arrangement. Local (greater Chicago area) students with no corporate site affiliation must take exams at Rice, Moffett or main campus.

   Internet students who have a schedule conflict with another live course must take the Internet course exam at a prescribed time that is immediately adjacent to or that overlaps with the scheduled course exam time. For example if a student registers in for CS 521-251 on the Internet and CS 458-051 live, there is final exam conflict because both courses meet on Tuesday nights. The official final exam date and time would be December 11th, 7:30-9:30pm for both courses. As a result, IIT Online Support Services might schedule the alternate CS 535-251 exam time with the instructor for 5:45-7:45pm for those students with a conflict. Students with a conflict must take exams at this time, unless otherwise indicated by the instructor.

   c. Students requesting lectures for the entire semester are required first to contact the course instructor for approval. This service is only available pending instructor approval. These students are also required to adhere to the same due dates and deadlines as the live course without exception. There is a fee for this service; students should contact Chuck Scott at (x75217 or scott@iit.edu).
4. IIT Online Technical and Support Responsibilities for Internet Courses -

   a. IIT Online Technical Services will capture and store live lectures, for many courses prepare class materials from instructor’s notes, handouts, materials provided by faculty before or immediately after the class, and video frames for posting.

   b. Technical support will be provided during the class for the TV broadcast. Additionally, support for the Internet course will be provided throughout the semester for both students and faculty. Trouble tickets may be submitted through the OTS Support Desk at http://support.iit.edu or 312-567-3375.

   c. IIT Online Support Services will provide proctors at all locations other than the origination site for all Internet students enrolled in the course. In some cases, this will involve two exam times to accommodate students who have a time conflict with another live and/or televised course.

   d. Internet courses originating at the Main Campus that overflow the room capacity during exam periods due to an influx of Internet students taking exams with the live section will require increased space in some cases. Larger rooms will be scheduled if possible, however in some cases our IIT Online may need to schedule a second room, as well as an additional proctor to accommodate the additional students.
# Illinois Institute of Technology

## IIT ONLINE Exam Schedule

**Course:**

**Instructor:**

### Exam 1

<table>
<thead>
<tr>
<th>Date</th>
<th>Time to begin</th>
<th>Time to end</th>
</tr>
</thead>
</table>

Check which are applicable:

- [o] In class
- [o] Open book
- [o] Open notes
- [o] Take home
- [o] Closed book
- [o] Closed notes

Will you be present during the exam? Yes/No

Special instructions:

Site (only if other than regular viewing site)

### Exam 2

<table>
<thead>
<tr>
<th>Date</th>
<th>Time to begin</th>
<th>Time to end</th>
</tr>
</thead>
</table>

Check which are applicable:

- [o] In class
- [o] Open book
- [o] Open notes
- [o] Take home
- [o] Closed book
- [o] Closed notes

Will you be present during the exam? Yes/No

Special instructions:

Site (only if other than regular viewing site)

### Exam 3

<table>
<thead>
<tr>
<th>Date</th>
<th>Time to begin</th>
<th>Time to end</th>
</tr>
</thead>
</table>

Check which are applicable:

- [o] In class
- [o] Open book
- [o] Open notes
- [o] Take home
- [o] Closed book
- [o] Closed notes

Will you be present during the exam? Yes/No

Special instructions:

Site (only if other than regular viewing site)

### Final Exam

<table>
<thead>
<tr>
<th>Date</th>
<th>Time to begin</th>
<th>Time to end</th>
</tr>
</thead>
</table>

Check which are applicable:

- [o] In class
- [o] Open book
- [o] Open notes
- [o] Take home
- [o] Closed book
- [o] Closed notes

If no exams are to be scheduled, please check here o

If you require students to come to an IIT campus, please check here o

**Faculty signature:** __________________________  **Date:** _________

If there are changes to this schedule, please notify Chuck Scott x75217 as soon as possible. Exams must be received at IIT ONLINE no less than 48 business hours prior to the exam date.
Illinois Institute of Technology  
IIT ONLINE Faculty Information Form

Instructor:  
Course(s):  
Semester:

The following information will provide IIT Online staff with contact information, and separately, for posting on the web for student use. If you do not wish us to post a particular contact type for the students, just enter “None” in the space provided, or if they may use that which you are providing for the IIT Online staff to use, you may just enter “Same.” Thank you.

Phone number: ___________________

    Phone Number for Students to Call: __________________________

Fax: __________________________

    Fax Number for Students to Use: __________________________

Email: ________________________

    Email for Student to Use: __________________________

Campus or preferred address: __________________________

    Mail address for Student to Use: __________________________

    URL to a specific departmental page or your home web page for Student Use: ____________________

Other: ____________________________________________________________________

RETURN to LIU YANG, IIT ONLINE, 103 STUART BUILDING or FAX to x75913
Logging to Blackboard
IIT Online Internet courses are delivered via Blackboard, IIT’s course management system. You can login to Blackboard via the myIIT portal at http://my.iit.edu. Instructions for logging-in are available at portal homepage. Once in Blackboard, under “My Courses”, click the respective course to access its site.

Viewing online lectures and materials
The course site contains several content areas including Course Information, Assignments, IIT Online, and more. IIT Online will post only the course videos and, if available, slides of materials shown in the live class session as a .PDF file within the IIT Online tab. (See the faculty packet for additional information.) All other material postings are the responsibility of the faculty. To access the videos and IIT Online prepared materials, click the IIT Online tab. A list of links to the course videos (camera icons), and accompanying .PDF files with printable slides, if available, will be shown. You will need to provide your UID (as username) and CWID (as password) to view course videos. Please note that all online materials are removed from Blackboard on the Friday after final exam week. All other content is set as “hidden.”

Adding content
Several course site areas, including Course Information, Course Documents, Assignments, and External Links, can host various files, links to websites, and other content.

To add a new file, click the Control Panel button at the bottom of the site menu. Then click one of the content area links at the top-left of the control panel page e.g. Course Information.

3 On the Add Item page, enter a name for the content item you are about to add in the Name field e.g. Course Syllabus. If desired, you can enter a description of the content item in the Text field. Next, click the Browse button located next to the Attach local file field, which appears below the Text field. Navigate to the file on your computer which you would like to upload. You cannot select more than one file at this step.

4 If desired, you can specify more options for this content item, such as availability dates.

5 After entering the necessary information for the item, scroll down to the bottom of the page and click the Submit button. Depending on your connection speed and specified file, it may take some time to upload it.

6 A confirmation page will be displayed when the content item has been added successfully. Click OK to return to the Course Information page, where the newly-uploaded item will appear. Buttons for modifying or removing this content item appear next to the item.

Viewing grades
To view the grades of the enrolled students, go to the Control Panel and click Gradebook under the Assessment area on the right.

Viewing enrolled students
1 To view a list of the currently (and previously) enrolled students, go to the Control Panel and click List/Modify Users under the User Management area.

2 To see a list of all users, leave the search field blank and click Search. Currently enrolled students will be listed in black text, while previously enrolled students who have left the course will appear grayed out.

Sending e-mail
1 To send an e-mail to all currently enrolled students, go to the Control Panel and click Send Email under the Course Tools area.

2 A list of options will appear, including All Users, Select Users, etc. Click All Users.

3 Type the message subject and body, attach a file if you like, and finally, click the Submit button.

Learning more about Blackboard
If you would like to learn more about Blackboard, you can refer to the Instructor Manual available from the Control Panel or the “Blackboard Faculty Manual” course within Blackboard itself (check “My Courses”). If you have a Blackboard issue or question, contact the OTS Support Desk by phone at (312)567-3375 or by email at supportdesk@iit.edu.