

## Math 350 – Introduction to Computational Mathematics

**Course Description from Bulletin:** Study and design of mathematical models for the numerical solution of scientific problems. This includes numerical methods for the solution of linear and nonlinear systems, basic data fitting problems, and ordinary differential equations. Robustness, accuracy, and speed of convergence of algorithms will be investigated including the basics of computer arithmetic and round-off errors. Same as MMAE 350. (3-0-3).

**Enrollment:** Required for AM and elective for other majors.

**Textbook(s):** Cleve Moler, *Numerical Computing with MATLAB*, SIAM.  
S. C. Chapra & R. P. Canale, *Numerical Methods for Engineers*, 5th Edition, McGraw Hill, 2006.

**Other required material:** Matlab or Maple

**Prerequisites:** Calculus, Differential Equations, basic Linear Algebra as acquired in MATH251, MATH 252, MATH 332 or MATH 333, and CS 105 or CS 115, or consent of instructor

### Objectives:

1. Students should gain an appreciation for the role of computers in mathematics, science and engineering as a complement to analytical and experimental approaches.
2. Students should have a basic knowledge of numerical approximation techniques, know how, why, and when these techniques can be expected to work, and have ability to program simple numerical algorithms in Matlab or other programming environments.
3. Students should have learned what computational mathematics is about: designing algorithms to solve scientific problems that cannot be solved exactly; investigating the robustness and the accuracy of the algorithms and/or how fast the numerical results from the algorithms converge to the true solutions. This includes a basic understanding of computer arithmetic and round-off errors and how to avoid loss of significance in numerical computations.
4. Students should be able to use and evaluate alternative numerical methods for the solution of linear and nonlinear systems, basic data fitting problems, and ordinary differential equations.
5. Students should be able to make appropriate assumptions to come up with a mathematical model that accurately reflects an appropriate scientific theory, and that is amenable to solution with a computer.
6. Students should appreciate the importance of written and graphical communication.

**Lecture schedule:** Two 75-minute (or three 50-minute) lectures per week

Course Outline:	Hours
1. Introduction to Computational Mathematics	10

<ul style="list-style-type: none"> <li>• mathematical modeling</li> <li>• review of Taylor series</li> <li>• numerical error (floating-point representation, computer arithmetic, round-off errors, and loss of significance in numerical computations)</li> <li>• programming in Matlab</li> </ul>	
2. Locating Roots of Equations	6
<ul style="list-style-type: none"> <li>• bisection method</li> <li>• Newton's method</li> <li>• secant method</li> <li>• introduction to the solution of systems of nonlinear equations - Newton's method for systems</li> </ul>	
3. Solving Systems of Linear Equations	6
<ul style="list-style-type: none"> <li>• direct methods (LU factorization)</li> <li>• basic iterative methods (Jacobi, Gauss-Seidel and SOR)</li> </ul>	
4. Interpolation	6
<ul style="list-style-type: none"> <li>• polynomial interpolation</li> <li>• piecewise polynomial and spline interpolation</li> </ul>	
5. Numerical Integration	4
<ul style="list-style-type: none"> <li>• Newton-Cotes methods</li> <li>• adaptive quadrature</li> </ul>	
6. Numerical differentiation and solution of ordinary differential equations	10
<ul style="list-style-type: none"> <li>• finite differences</li> <li>• Runge-Kutta methods</li> <li>• multistep methods and stiff equations (comparison of various Matlab stiff solvers)</li> <li>• FFT and spectral methods</li> </ul>	

<b>Assessment:</b>	Homework	10-30%
	Computer Programs/Project	10-20%
	Quizzes/Tests	20-50%
	Final Exam	30-50%

**Syllabus prepared by:** Greg Fasshauer and Dietmar Rempfer

**Date:** Oct.13, 2006