Mathematics 5610
Introduction to Numerical Analysis I
Fall 2005 (MTWF 8:35-9:25, LCB 225)

Instructor: Grady Wright          Phone: 581-8649
Office: JWB 126                  E-mail: wright@math.utah.edu
Office Hours: M,F 9:25am-11:35am, or by appointment


Website: http://www.math.utah.edu/~wright/courses/5610

Course description: This is the first half of a year long introductory course on numerical analysis at the 5000 level. The topics covered this semester include numerical methods for linear and nonlinear equations, interpolation, approximation theory, numerical integration and differentiation. If time permits, we may also cover eigenvalues and eigenvectors. In the second semester (MATH 5620), we cover more advanced numerical linear algebra topics (e.g. over/underdetermined linear systems, singular value decomposition), fast Fourier transform (FFT), numerical methods for solving ordinary differential equations (ODEs) (both initial-value and boundary-value problems) and partial differential equations (PDEs).

Prerequisites: No previous experience in numerical analysis is necessary. However, knowledge of the following topics is required:

• Single and multivariable calculus (MATH 2210)
• Ordinary differential equations and linear algebra (MATH 2250 or 2270)
• Computer programming (using, for example, C/C++, Java, Fortran, or MATLAB)

Computational and Engineering Science: This class fulfills one of the requirements for the Computational Engineering and Science (CES) Program here at the University. Enrollment in the program is necessary to obtain CES Certificate or MS credit. If you are interested in learning more about the CES program, please visit http://www.ces.utah.edu or contact Coralee Bernard (coralee@cs.utah.edu, 581-3455).

Homework: Homework is handed out in class (and posted on the class website) every other Tuesday. The problems involve a mix of theory and computing. Regarding the latter, please read the text below the Programming heading.

Your submitted homework should show all necessary work you used to solve the problems; mathematical statements should be complete (or nearly complete) sentences; and the reasoning and logic underlying all arguments should be clearly spelled out. Computer programs should be included along with numerical results presented in a readable format (e.g., in a table with headings or in a plot with labels). Please see the document “How to Present your Work” on the course web page for tips on meeting these requirements. Failure to adhere to the above requirements may result in a loss of points.

Homework assignments are due on Tuesday at the end of class. Late homework assignments are accepted up to three days beyond their due date. However, a 10% penalty is applied for each twenty four hour period they are late. This means that if an assignment is turned in after 9:25am on Tuesday, but before 9:25am on Wednesday, then 10% is automatically deducted. Late homework assignments need to
be put into my box (located JWB 228) or, if you have the means, e-mailed to me. **Please indicate the date and time you put the homework in my box or I will assume it was turned in moments before I retrieved it from my box.**

**Teamwork:** Teamwork is part of the real world and therefore permitted (and encouraged) for all homework assignments (but not the exams). **Keep in mind that the purpose of teamwork is to enhance the learning effect, not to decrease the workload.** It is up to each team member to prevent abuse. Please observe the following rules:

- **No teams with more than three members!**
- If you work in a team then clearly state so and hand in just one set of answers. If you collaborate on some problems but not on others then **clearly state so** on your turned-in solution.
- In general, you can consult literature and people, but **you have to acknowledge all help so obtained** (except for the textbook or myself).

Homework solutions (taken from selected students' solutions) are posted on the course website once the grading has concluded.

**Grading Policy:** The final grade for the course is based on homework assignments, a midterm exam, and a final exam. The breakdown for the course grade is as follows:

- Homework and midterm: 70%
- Final (Thursday, Dec. 15, 2005 8:00-10:00am): 30%

The midterm counts as two homework assignments. The lowest homework score is discarded (half of the midterm can be discarded).

**Programming:** I highly recommend that you use MATLAB for all programming assignments involving numerical computations. An important part of numerical analysis today is the use of commercial or public domain software packages for solving particular problems. In order to gain exposure to this side of numerical analysis, students need to practice using such packages. MATLAB offers the perfect opportunity for such practice; it is one of the most dominant commercial computing environments. The MATLAB language is intended to be easy to learn and use, while still being extremely powerful. Three other reasons for using MATLAB for the programming assignments is that: 1) The webpage for the book contains a MATLAB implementation of several algorithms discussed; 2) My examples and homework assignments will use MATLAB code; 3) I will help you debug your programs.

There is a freeware “clone” of MATLAB called Octave ([http://www.octave.org](http://www.octave.org)) that is also suitable for doing your programming assignments. However, note that Octave does not contain all the wonderful functionality of MATLAB.

If you have not used MATLAB previously, help resources are available on the course website and from the instructor. MATLAB is available in the Undergraduate and Graduate student mathematics computing labs (as well as various other locations on campus). For more information on these labs, including hours of operation and location, go to [http://www.math.utah.edu/ugrad/lab/](http://www.math.utah.edu/ugrad/lab/). You can also purchase a student version of MATLAB for a heavily discounted price from the book store.

Problems involving symbolic computations can be done using Maple or Mathematica. The former is available in the Undergraduate and Graduate student mathematics computing labs.

**Important Dates:**
• Sept. 2 – Last day to drop term and first session classes (and get your money back)
• Sept. 5 – Labor Day – No Class
• Sept. 6 – Last day to add classes and to elect CR/NC or audit for term and first session
• Oct. 6-7 – Fall break – No class
• Oct. 21 – Last day to withdraw from term length classes
• Dec. 2 – Last day to reverse CR/NC for term and second session
• Nov. 24-25 – Thanksgiving break – No Class
• Dec. 8 – Classes end
• Dec. 15 – Final exam, 8:00-10:00am

Material to Cover: We will cover the following material (in roughly this order):

1. Introductory material (Ch. 1)
2. Linear equations (Ch. 3.0 – 3.9)
3. Nonlinear equations (Ch. 2.1 – 2.7, 3.10)
4. Interpolation and Approximation (Ch. 5 and my notes)
5. Numerical differentiation and integration (Ch. 6)
6. Eigenvalues and Eigenvectors (Ch. 4) – if time permits

References:
• Atkinson, An Introduction to Numerical Analysis (John Wiley and Sons)
• Dahlquist and Björck, Numerical Methods (Dover reprint 2003)
• Henrici, Essentials of Numerical Analysis (John Wiley and Sons)
• Fröberg, Introduction to Numerical Analysis (Addison-Wesley)
• Trefethen and Bau, Numerical linear algebra (SIAM)
• Golub and Van Loan, Matrix Computations (Johns Hopkins University Press)
• Demmel, Applied Numerical Linear Algebra (SIAM)
• Fornberg, A Practical Guide to Pseudospectral methods (Cambridge University Press)

The purpose of computing is insight, not numbers! –Richard Hamming

It would appear that we have reached the limits of what it is possible to achieve with computer technology, although one should be careful with such statements, as they tend to sound pretty silly in 5 years. – John Von Neumann (1949)

The badly named real number system is one of the triumphs of the human mind. It underlies the calculus and higher analysis to such a degree that we may forget how impossible it is to deal with real numbers in the real world of finite computers. But, however much the real number system simplifies analysis, practical computing must do without it. – George Forsythe

Without precise calculations we could fly right through a star or bounce too close to a supernova and that’d end your trip real quick, wouldn’t it? –Hans Solo