Course Outline for ENGR 308: Computer Methods in Engineering

Elective
Civil Engineering and Mechanical Engineering

Bulletin Description
ENGR 308 Computer Methods in Engineering (3 units)
Prerequisite: ENGR 103, ENGR 201 or ENGR 305, and MATH 245.
Methods of numerical analysis for engineering problems; development of computer
programs for solution of problems drawn from various branches of engineering; analysis
of error propagation in computations.

Textbook

Coordinator
V.V. Krishnan, Professor of Mechanical Engineering

Prerequisite by Topic
1. Knowledge of Programming in an Accepted Computer Language
2. Differentiation, Integration, Power Series
3. Basic Concepts of Linear Algebra
4. Understanding of the Design Process

Course Objectives
1. To provide a working knowledge of commonly used methods of numerical analysis in
   the solution of engineering and scientific problems. [A.1]
2. To enable the students to formulate practical engineering problems in terms
   amendable to computer solutions. [A.1, A.2, B.3]
3. To provide the students an opportunity to develop the necessary programming skills
   required for converting a feasible mathematical technique into a working computer
   method. [A.1, A.2, A.5, B.4]

---

1 Numbers in parentheses refer to the educational goals and objectives of the School of Engineering
Topics
1. Floating point arithmetic, error propagation, numerical instabilities
2. Solution of nonlinear equations
3. Solution of simultaneous linear equations
4. Numerical differentiation and integration
5. Solution of differential equations
6. Interpolation and curve-fitting methods

Professional Component
1. Engineering Science 33%
2. Engineering Design 33%
3. Mathematics 33%

Evaluation
1. Exams
2. Short quizzes
3. Design-oriented programming problem assignments
4. Periodic homework problems

Performance Criteria²
Students will have:

Objective 1
1.1 Knowledge of the principles and operation of various numerical methods. [1, 2, 4]
1.2 Ability to determine the appropriate solution methods for a particular classes of problems and situations. [1, 2, 3, 4]

Objective 2
2.1 Ability to formulate specific problem types in terms of required computer routines. [3, 4]
2.2 Ability to identify and use appropriate software. [3, 4]

Objective 3
3.1 Ability to write a program segment in a standard programming language. [3]
3.2 Ability to document and/or debug programs. [3]

Spring Semester, 2005
Instructor: Michael Strange
Office: SCI 140
Phone: 415/405-2419
E-mail: strange@sfsu.edu

² Numbers in brackets represent the evaluation tools used in assessing student performance
Class/Laboratory Schedule
Two 75-minute lecture sessions/week

Prepared by
Michael Strange, Spring, 2005