BEE151. Introduction to Computer Programming Fall 2007

Catalogue description: Fall. 4 credits. Prerequisite: MATH 191 or equivalent (coregistration permissible). Each lab and recitation section limited to 22 students.

An introduction to computer programming and concepts of problem analysis, algorithm development, and data structure in an engineering context. The programming language MATLAB with Simulink is used, implemented on personal computers, and applied to problems of interest in Biological and Environmental Engineering. No previous programming experience is assumed although basic computer skills are useful (e.g., word processing, email).

Required or elective: Required (in BEE)

Textbook(s) and other materials:
Custom textbook consisting of chapters from the following texts:


Course objectives:
BEE 151 has several objectives. The first is to learn how to analyze and then write computer programs to solve real-life problems. The language used is Matlab, and the computers are Windows 2000-based. A second objective is to use problems for laboratory and homework exercises that reflect aspects of professional work done by Biological and Environmental Engineers. A third objective is to learn how structure and planning can lead to solving engineering-type problems efficiently and to recognize and write computer programs which are user-friendly, technically correct, logically based, and well documented (this is, perhaps, the most important objective). A fourth is to have, in spite of the work, an occasional fragment of fun while learning.

Topics covered:
Structure of Matlab Programs
Algorithm Creation and Structure
Reserved Words, Identifiers, Data Types, etc.
Assignment Statements and script files
Functions
Conditional and repetitive control structures
Arrays
Files and I/O
Graphical User Interfaces
Sorting and Recursion
Simulink and engineering system modeling

Class/laboratory schedule:
Three 50-minute lectures per week and 2-hour lab per week

Course outcomes and their relation to ABET program outcomes a-m:
1. Be fluent in the use of procedural statements—assignments, conditional statements, loops, function calls—arrays, and recursion. Be able to design, code, and test MATLAB programs that meet requirements expressed in English. This includes a basic understanding of top-down design. (a, c, k)
2. Have knowledge of solution techniques for differential equations in Matlab. (a, k)
3. Have a working familiarity with graphics tools in MATLAB. (k)
4. Have used a design process in using the computer to solve engineering problems. (e)
5. Practice effective written technical communication skills. (g)
Assessment of course outcomes:
Assessment is based on five case studies (four as homework exercises, one as a term project), three one-hour exams, a course journal, and a final examination of 2.5 hours.

Person preparing this description and date:
C. Lindsay Anderson, 12/18/07

Ethical behavior statement:
The dark side of shared work is never a pleasant subject. Talking about your work with others can help the educational process a great deal, and such interactions are encouraged. However, when the computer keys are pressed and the program is written, the code should be yours, and yours alone. Programming is such an individualized process that sharing another's work becomes quickly obvious and will be challenged.

The following statement is from the University's Code of Academic Integrity:
"When writing a program assignment, a student may discuss general strategies to be employed and perhaps receive some help in learning how to test the program to find errors, but unless closer cooperation is expressly permitted on the assignment, the actual writing of the program and its detailed testing must be the work of the individual student. Any other assistance should be expressly acknowledged."