



## **GEU 111 Engineering Problem Solving and Computation** Spring 2007

### **Instructor:**

Susan Freeman, 359 SN (mailbox in 334 SN), 373-4954  
IM screen name: fishirtwife

**email:** [sfreeman@coe.neu.edu](mailto:sfreeman@coe.neu.edu)  
See Blackboard for course info

### **Conference Hours: tentative...**

Tuesday 3:00-4:30 PM in 429 Dana

Wednesday 1:00-2:00 PM in my office and Thursday 2:50 - 3:30, or other times by appointment

Tuesday evening online 8:00-10:00 PM and during office hours and when available.

### **Required Materials:**

- ▽ **Applications Programming in C++**, Johnsonbaugh and Kalin, Prentice-Hall, 1999
- ▽ **Matlab: An Introduction with Applications**, 2<sup>nd</sup> Edition, A. Gilat, Wiley, 2005.

### **References** (on reserve in Snell Library):

- ▽ C++, How to Program, 2nd Edition, Deitel & Deitel, Prentice-Hall, 1998
- ▽ CORE C++, A Software Engineering Approach, Victor Stern, Prentice-Hall, 2000
- ▽ Several other C++ references are in the reference section of the library and on the shelves.

### **Course Goals:**

- (1) develop skills in algorithmic thinking by preparing computer programs to analyze and present engineering data
- (2) properly evaluate and interpret the results of programming work
- (3) learn to write (in C++ and Matlab) the types of programs needed for engineering problem solving
- (4) become familiar with a wide variety of interesting and challenging engineering problems that can be solved with programmable algorithms, and

### **Course Objectives:**

- (1) Create flow charts that illustrate algorithmic thinking. Use flow charts in preparing to write C++ and Matlab programs.
- (2) Follow and document each step in the Problem Solving Method (input, output, solve by hand, scientific principles, convert to code, test code).
- (3) Use the programming language C++ and Matlab to write computer programs that solve engineering problems.
- (4) Use data types, input/output commands, loops, control structures, functions, arrays, pointers and other programming language constructs in C++ computer programs.
- (5) Write cover letters that describe succinctly how the problem was solved.

**Computer Background.** No prior programming experience is assumed, except for those techniques learned in GE U110.

**Hardware/Software.** The course will be conducted assuming that students will use the software on NUNET. C++ packages can be purchased at the bookstore and in many other places.



### **Grading Formula:**

10% attendance/daily assignments, 35% weekly assignments, 35% weekly quizzes, and 20% final project.

### **Assignments and Attendance:**

**A.** There will some **daily** assignments due. They will be graded on a 3-point scale:

3 = you did the whole assignment (not necessarily correctly, but completely)

2 = you made an attempt

1 = you turned in a blank piece of paper with your name to get credit for attending class

0 = you didn't attend class

**Late daily assignments will not be accepted** since we will be discussing them in class. In averaging your score for daily assignments, the lowest grade will be dropped.

**B.** There will be **weekly** (or so) homework sets requiring use of the computer and problem solving skills. They are due at the beginning of class on the date due. **Late weekly assignments will incur penalties as determined by the instructor.** Generally, 10% will be deducted for each day, assignments over a week late will NOT be accepted. In averaging your score for weekly assignments, no grades will be dropped. **NOTE:** All assignments must be presented in a neat, organized, and professional manner; assignments presented otherwise are returned with penalty.

**C.** In partial fulfillment of the student portfolio requirements, each student will submit (at least) one cover letter with an assignment, as designated by the instructor. Following its review by the instructor, this cover letter should be included in the student's portfolio. We also recommend that one or more programming projects / assignments be included in this portfolio as an example of the student's work in this course.

**Disabilities:** If you have specific physical, psychiatric, or learning disabilities and require accommodations, please let me know early in the semester so that your learning needs may be appropriately met. You will need to provide documentation of your disability to the Disability Resource Center, located in 20 Dodge Hall.

### **Academic Honesty:**

Plagiarism and cheating will not be tolerated; they will be dealt in accordance with University policies described in the Student Handbook. All engineering majors should be familiar with the Honor Code of our College of Engineering that is included in the GEU 100 course material, and with professional engineering codes of ethics (see, for example, the NSPE Code of Ethics presented in the Engineering by Design textbook on pages 511-514). Students may be working on assignments together, and submitting their teams' work, but, submission of other teams' work as your own is not permitted. **Copying of computer code is not allowed.** Each student/team is expected to prepare and submit their own programs, reports, drawings, and other materials. If two student/teams' work is suspiciously similar, a penalty may be assessed. Violators will be referred to the Student Court for review, where penalties may include but are not restricted to: zero credit on the work, student placed on probation, and submission of information on judgment in the student's permanent record.

Good Luck, Work Hard, and Enjoy!



**Material to be covered:** JK = Applications Programming in C++, Gilat = Matlab

**Week 1** (1/8-11) Read Chapter 0 and Chapter 1 - JK

Topics: Computer Organization, number representations, assembly, compilation, Why C++?  
Algorithm development, flow charting, directing program flow.

**Week 2** (1/17-18) Read Chapter 2 - JK

Topics: Data types, operators, some commands (Chapter 1) and start Control flow

**Week 3** (1/22-25) Read Chapters 2 and start Chapter 3 - JK

Topics: More Control flow and operators, formatting

**Week 4** (1/29-2/1) Chapter 3 - JK

Topics: Functions, Arguments and parameters, Scope of variables

**Week 5** (2/5-8) Chapter 3 - JK

Topics: More Functions (practice)

**Week 6** (2/12-15) Read Chapter 4.1 – 4.2, 4.4 – 4.8 - JK

Topics: Arrays; character strings, Arrays in Functions

**Week 7** (2/21-22) Read Chapter 4.9 – 4.10 - JK

Topics: 2-D Arrays

**Week 8** (2/26-3/1) Chapters 0-4

Topics: Arrays and C++ Exam

**Week** (3/5-9) Spring Break

**Week 9** (3/12-15) Read Chapters 1, 2, 4.1, 4.3, 5.1, 5.5 Gilat

Using Matlab, Arrays, Operations with Arrays, Script files, Plotting

**Week 10** (3/19-22) Read Chapter 3, 4.4 – 4.7 Gilat

Matrix Operations, Linear Equations, File Formatting

**Week 11** (3/26-3/29) Read Chapter 6, 7 Gilat

Functions and Program Control

**Week 12** (4/2-5) Read Chapter 10.1 Gilat

Numerical Integration, review, other...

**Week 13** (4/9-4/12) Combining Matlab and C++....

**Week 14** (4/18) Review and Projects, final quiz

(4/20 and 4/23-4/27) Final Exam Days



## COURSE TOPICS

### **Problem Solving – every week**

- Problem Definition (Outputs, Inputs, Diagram)
- Flow Chart or Pseudocode
- Testing (Hand Calcs)
- Coding
- Debugging (Verification and Validation)
- Summarizing (memo)

### **C++ and Matlab– add a little more each week**

- Program Structure
- Variables and Constants
- Keyboard Input and Screen / File Output
- Mathematical Operations – order of, mixed
- if statements
- Loops
- Input Files
- Functions – multiple inputs, single output
- Arrays
- Functions with arrays
- Pointers (functions with multiple outputs)
- Plotting – 2D and 3D