16.216: ECE Application Programming
Spring 2013

Syllabus

Course Meetings

Section 201: MWF 12-12:50, Ball 326

Course Website

Main page: http://mgeiger.eng.uml.edu/16216/sp13

Schedule: http://mgeiger.eng.uml.edu/16216/sp13/schedule.htm

Course Discussion Group

All course announcements will be posted on the discussion group—you are responsible for checking the board regularly or enabling direct e-mail updates from Piazza.

Sign up link: http://piazza.com/uml/spring2013/16216

Instructor

Dr. Michael Geiger
E-mail: Michael_Geiger@uml.edu
Office: Perry Hall 118A
Phone: 978-934-3618 (x3618 on campus)
Office hours: TBD *(likely Monday 1-2:30, Wednesday 1-2:30, Thursday 1-3)*

Once finalized, my office hours are the minimum that I will be available—I will (usually) be on campus five days a week. Feel free to stop by my office, e-mail me questions, or schedule a one-on-one appointment. Office hours are subject to change.

Textbook

K.N. King, *C Programming: A Modern Approach*, 2nd edition, 2008, W.W. Norton.
ISBN: 978-0-393-97950-3

Course Overview

Catalog Description: Introduces C programming for engineers. Covers fundamentals of procedural programming with applications in electrical and computer engineering and embedded systems. Topics include variables, expressions and statements, console input/output, modularization and functions, arrays, pointers and strings, algorithms, structures, and file input/output. Introduces working with C at the bit manipulation level. Laboratories include designing and programming engineering applications.

Credits: 3

Prerequisites: 25.108 (Intro to Engineering II) and ECE major

Course Overview (cont.)

Course Objectives: By the end of this course, you should understand and be able to use
all of the following:

1. **Basic C Language Concepts:** constants, variables, operators, expressions and
   assignment statements
2. **Input and Output:** Reading data from the keyboard and displaying formatted
   results on the screen
3. **Flow of Control 1 – Decisions and selection:** `if` and `switch` statements
4. **Flow of Control 2 – Repetition:** `while`, `do-while`, and `for` loops
5. **Functions**: Defining and calling functions. Using arguments to pass data to a
   function. Using arguments to obtain results from a function. Return values.
6. **Data Structures 1**: One and two-dimensional arrays. Character strings.
7. **Data Structures 2**: Structures, collections of data components of differing types.
8. **File Input / Output**: Writing programs which obtain input from a file rather than
   the keyboard, and which write results to a file rather than to the screen

Grading: Grades will be computed on an A to F scale; no A+ grades will be assigned, in
accordance with UMass Lowell policy. The weights assigned to the various items are:

Programming assignments	60%
Exam 1	10%
Exam 2	15%
Exam 3	15%

Incomplete grades will only be given in exceptional situations, and the student must be
passing the class at the time the grade is requested.

Programming assignments: Typically, you will have one week to complete each
assignment. All assignments will be graded according to the program grading guidelines,
to be distributed separately. Late assignments will lose 2^{n-1} points per day, including
weekends and holidays. You will submit your work via e-mail directly to Dr. Geiger.

For each assignment, you will be allowed one resubmission to improve your grade
without penalty. You must resubmit your code by the given deadline for that assignment;
late penalties will apply to late resubmissions. Note that the resubmission policy does not
allow you to avoid penalties when the original submission is late (e.g., an assignment
losing 4 points for a late initial submission has a maximum possible score of 96 for the
resubmission). See the grading guidelines for more details.

Exams: Make-up exams will only be offered in exceptional circumstances. You must
notify Dr. Geiger as early as possible in order to determine an appropriate make-up date.

Class participation: You are responsible for all material discussed or announced in class.
You are expected to attend class regularly and participate in any in-class discussions, as
such exercises are essential to your learning. Although lecture attendance is not explicitly
required, regular attendance will improve your understanding of the course concepts.

Academic Honesty

All assignments and exams must be completed individually unless otherwise specified. You may discuss concepts or material covered in class, but may not share any details of your solutions to assigned problems, including algorithms and code. Plagiarism (in this course, copying code from an outside source) is also unacceptable and will be treated as an instance of cheating.

Students are allowed to discuss assignments in general terms and to help one another fix specific errors—examples include compiler errors or output formatting. In this case, students are required to note that they received assistance from a classmate by listing that person's name and the nature of their assistance as part of their assignment header. However, any sharing of code—even when used strictly to help a classmate solve a specific error—is a violation of the academic honesty policy.

Any assignment or portion of an assignment that violates this policy will receive a grade of zero for all parties concerned. Depending on the severity of the infraction, or in cases of repeat violations, additional penalties may be given at the instructor's discretion, up to and including a failing grade in the course.

Note that all instances of academic dishonesty will be reported to the Provost's Office, as required by the University's Academic Integrity Policy. Further information on this policy can be found at:

http://www.uml.edu/catalog/undergraduate/policies/academic_dishonesty.htm

Course Schedule

This schedule contains a tentative schedule of topics we will cover throughout the term; the course website will contain the most up-to-date version. The web page will also describe which section(s) of the textbook are associated with each lecture.

Please note that several days are denoted as "PE#"—in these classes, we will do an in-class programming exercise. While students will be able to participate even if they do not have a computer, I suggest anyone with a laptop bring it to class on these days.

Please note that the exam dates are fixed—the first exam will be held on **Wednesday, February 20 in class**, the second exam will be held on **Wednesday, March 27 in class**, and the third exam will be held **on Thursday, May 9, from 8:00-11:00 AM**.

Week	Date (M)	Lecture Topics	Suggested Reading	Programs
1	1/21	*No Mon. lecture--Martin Luther King Day* 1. Course introduction/overview; C programming cycle 2. Basic C program structure; data in C	W: None F: 2.1-3, 2.8	Program 1 (due 1/28)
2	1/28	3. Variables (+ using printf() with variables) 4. Input with scanf(); printf() formatting 5. PE1 (Flowcharts, debugging basics)	M: 2.4, 2.6-7, 3.1, 7.1-3 W: 2.5, 3.2 F: None	Program 2 (due 2/4)
3	2/4	6. Operators; conditional statements: if 7. Conditional statements: if/switch 8. Finish conditional statements	M: 4.1-5, 5.1-2 W: 5.3 F: None	Program 3 (due 2/11)
4	2/11	9. Loops: while, do-while 10. Loops: for 11. PE2 (Conditionals, loops)	M: 6.1-2 W: 6.3-4 F: None	Program 4 (due 2/25)
5	2/18	*No Monday lecture--Presidents Day* 12. Exam 1 Preview *(Tuesday, 2/19)* **Wednesday, 2/20: EXAM 1** 13. Exam 1 Review	No new reading	
6	2/25	14. Functions 15. Function examples; pointers 16. Pointer arguments; fn./pointer examples	M: 9.1-4 W: 11.1-5 F: None	Program 5 (due 3/4)
7	3/4	17. PE3 (Functions) 18. Math library; One dimensional arrays 19. 2D arrays; arrays and functions	M: None W: 23.3, 8.1 F: 8.2, 12.1-4	Program 6 (due 3/20)
8	3/11	*Spring Break--no classes*		

Course Schedule (cont.)

Week	Date (M)	Lecture Topics		Programs
9	3/18	20. Two dimensional arrays and functions 21. Character arrays and strings 22. PE4 (Arrays and strings)	M: None W: 13.1-6 F: None	Program 7 (due 4/1)
10	3/25	23. Exam 2 Preview **Wednesday, 3/27: EXAM 2** 24. Exam 2 Review	No new reading	
11	4/1	25. File I/O 26. General I/O 27. PE5 (File I/O, general I/O)	M: 22.1-3 W: 22.4-5 F: None	Program 8 (due 4/8)
12	4/8	28. Structures *Wednesday, 4/10: Last day to withdraw* 29. Structures (continued) 30. PE6 (Structures)	M: 16.1-3 W: None F: None	Program 9 (due 4/15)
13	4/15	*No Monday lecture--Patriots Day* 31. Bitwise operators 32. Dynamic memory allocation	W: 20.1 F: 17.1-4	Program 10 (due 4/22)
14	4/22	33. Dynamic memory allocation (continued) 34. PE7: Dyn. allocation and data structures 35. Topics TBD	M: 17.5-6 W: None F: TBD	Program 11 (due 4/29)
15	4/29	36. Topics TBD 37. Exam 3 Preview *Classes end Wednesday, 5/1*	M: TBD W: None	
	5/9	**EXAM 3: THURS., 5/9, 8:00-11:00 AM**		