

Syllabus - CMSC 212 – Spring, 2007

Instructors:	
0101 & 0102	0201 & 0202
Jeff Hollingsworth	Jan Plane
4155 AV Williams	1113 AV Williams
301 405-2708	301 405-2754
hollings@cs.umd.edu	jplane@umd.edu

Discussion Section TAs:		
0101 & 0102	Troche, Jose	jjatroche@cs.umd.edu
0201 & 0202	Trapnell, Cole	cole@cs.umd.edu
Other TAs:		
Huynh, Thuan		thuan@cs.umd.edu
Thakor, Shashvat		shash@cs.umd.edu
Katrina LaCurts		klacurts@umd.edu

Class URL: <http://www.cs.umd.edu/class/spring2007/cmssc212>

You are expected to check the class web page on a regular basis (at least every couple of days).

Catalog Description:

This course introduces many of the concepts that lie behind software, such as hardware, memory layout, memory management, and operating systems. It explains how these concepts affect the design of software systems. This course provides a transition from the Java environment of the preceding two courses to programming in C.

Objective:

A good working knowledge of how to program in the C programming language. Understanding how write program with explicit memory allocation and de-allocation. An introduction of how to write systems programs.

Prerequisites:

CMSC 132 (with a grade of C or higher), Corequisite: CMSC 250 (or equivalent).

Topics Covered (in approximately the order we will cover them):

- Introduction: Moving from Java to C (2 1/2 weeks)
- Pointers and memory management (3 1/2 weeks)
- I/O (2 weeks)
- Libraries and linking (1 week)
- Program measurement (1 week)
- Primitive data representation (1 week)
- Array representation (2 weeks)
- Implementation of object-oriented concepts (1/2 week)
- Multi-language support (1/2 week)

Required Course Text: *Pointers on C, Reek*, Addison-Wesley 1998; ISBN 0-673-99986-6.

Recommended Course Text: *Computer Systems: A Programmer's Perspective*, Bryant and O'Hallaron, Prentice Hall 2003; ISBN 0-13-034074-X.

Programming Projects:

Understanding programming concepts is a hands-on activity. This class will include several substantial programming projects that will require students to read and understand provided code, write new modules, and debug the resulting system. The instructors reserve the right to **fail**, regardless of overall numeric score, students who do not submit a good faith attempt to complete all programming assignments.

Grading:

Final Exam	25%
Midterms (2 each worth 15%)	30%
Programming Assignments	35%
Quizzes (in discussion section)	10%

Exams:

Midterm #1 – Thursday, March 8, 6:00pm – 7:30pm

Midterm #2 – Thursday, April 10, 6:00pm – 7:30pm

Final – TBA (May be different from what is on the Schedule of Classes)

Schedule Conflicts: If you have a schedule conflict with any exam, please contact your instructor by the end of the schedule adjustment period (2/6/07).

Re-grade policy. All requests to change grading of homework, programming projects, or exams must be submitted via the grades web system within one week of when the results of the assignment were made available. Requests to re-grade exams or quizzes must also include the exam booklet to your discussion section TA by the deadline. Requests must be specific, convincing, and explain why you feel your answer deserves additional credit. A request to re-grade an assignment can result in the entire assignment being re-evaluated and as a result the score of *any part* of the assignment may be increased **or lowered** as appropriate.

Cell phones: As a courtesy to your fellow classmates, pagers and cell phones must be off or on vibrate during class, discussion section, and exams. Having cell phones or pagers ring during course meetings can result in points being deducted from your semester grade.

Illness: Requests to make up exams due to illness must be accompanied by a note from a medical doctor and include the doctor's phone number.

DSS: If you require any special accommodations, you must provide paperwork from DSS by the end of schedule adjustment period (2/6/07).

Academic Integrity:

All work that you submit in this course must be your own; unauthorized group efforts will be considered academic dishonesty. See the Undergraduate Catalog for definitions and sanctions. Academic dishonesty is a serious offense that may result in **suspension or expulsion** from the University. In addition to any other action taken, the grade "XF" denoting "failure due to academic dishonesty" will normally be recorded on the transcripts of students found responsible for acts of academic dishonesty. Sharing of code on programming assignments is a form of academic dishonesty.

Late Policy:

No late homework or programming assignments will be accepted. If you are unable to complete a programming assignment due to serious illness or family emergency, please see your instructor as soon as possible to make special arrangements.

Office Hours and Email:

We (the TAs and professors) are happy to answer questions during office hours, and by email. However, office hours and email are not intended as a replacement for lecture. As a result, we will only see people during office hours or respond to your email if you regularly attend class. Due to our own work schedules, we may not respond to email instantly. However, we will try to respond to your email by the next regularly scheduled office hour after you send it.