

Computer Organization & Assembly Language Programming

Instructor.

Prof. Richard Chang, chang@umbc.edu, 410-455-3093.

Office Hours: Tuesday & Thursday 10:00am–11:00am, ITE 326.

Teaching Assistants.

Roshan Ghumare, roshang1@umbc.edu

Genaro Hernandez, Jr., genaroh1@umbc.edu

Time and Place.

Section 01: Tu - Th 1:00pm–2:15pm, ITE 229.

Section 02: Tu - Th 2:30pm–3:45pm, ITE 229.

Textbook.

- *Essentials of Computer Organization and Architecture*, third edition, by Linda Null & Julia Lobur. Jones & Bartlett Learning, 2010. ISBN: 1449600069.
- *Assembly Language Step-by-Step: Programming with Linux*, third edition, by Jeff Duntemann. Wiley, 2009. ISBN: 0470497025.

Web Page. <http://umbc.edu/~chang/cs313/>

Catalog Description. This course introduces the student to the low-level abstraction of a computer system from a programmer's point of view, with an emphasis on low-level programming. Topics include data representation, assembly language programming, C programming, the process of compiling and linking, low-level memory management, exceptional control flow, and basic processor architecture.

Prerequisites. You should have mastered the material covered in the following courses: CMSC 202 Computer Science II and CMSC 203 Discrete Structures. You need the programming experience from CMSC202. Additional experience from CMSC341 341 Data Structures would also be helpful. Also, you must be familiar with and be able to work with truth tables, Boolean algebra and modular arithmetic.

Objectives. The purpose of this course is to introduce computer science majors to computing systems below that of a high-level programming language. The material covered can be broadly separated into the categories of assembly language programming, C programming and digital logic. These topics prepare the students to take CMSC411 Computer Architecture and CMSC421 Operating Systems which are required courses for the computer science major.

Under the heading of assembly language programming students will be introduced to the i386 instruction set, low-level programming, the Linux memory model, as well as the internal workings of compilers, assemblers and linkers.

C programming topics will concentrate on dynamic memory allocation.

Topics under computer organization include digital logic design (combinational circuits, sequential circuits, finite state machines) and basic computer architecture (system bus, memory hierarchy and input/output devices).

Grading. Your final grade will be based upon 5 homework assignments (15% total) and 8 programming assignments (40% total). There will also be a midterm exam (20%) and a final exam (25%). However, if some homework or programming assignments are canceled and not made up, the proportion of your grade from homework, projects and exams will remain the same. For example, if a programming assignment is canceled, then each programming assignment would be worth 5.714% (instead of 5%). That keeps programming assignments at 40% of your final grade.

Your final letter grade is based on the standard formula:

$$0 \leq F < 60, \quad 60 \leq D < 70, \quad 70 \leq C < 80, \quad 80 \leq B < 90, \quad 90 \leq A \leq 100$$

Depending upon the final distribution of grades in the class, there may be a curve in your favor, but under no circumstances will grades be curved downward.

Grades are given for work done *during* the semester; incomplete grades will only be given for medical illness or other dire circumstances.

Due Dates. There will be a homework or programming assignment due for every week of the class (except for the week of the midterm exam). Written homework assignments are due at the beginning of lecture. Programming assignments and logic simulations are submitted online. Electronic submissions are due at 11:59pm of the due date.

In general, late assignments will not be accepted. However, each student may submit one assignment (of any kind) up to one week late during the semester.

Academic Integrity. You are allowed to discuss the homework assignments with other students. However, circuit simulation exercises and programming projects must be completed by individual effort. (See the Project Policy.) Furthermore, you must write up your homework *independently*. This means you should only have the textbooks and your own notes in front of you when you write up your homework — not your friend's notes, your friend's homework or other reference material. You should not have a copy of someone else's homework or project *under any circumstance*. For example, you should not let someone turn in your homework.

The UMBC Undergraduate Student Academic Conduct Policy is available at:

http://www.umbc.edu/undergrad_ed/ai/documents/ACC2011.pdf

Exams. The exams will be closed-book and closed-notes. The midterm exam has been scheduled for Tuesday, October 23. For Section 01 (TuTh 1-2:15pm), the final exam is on Tuesday December 18, 1-3pm. For Section 02 (TuTh 2:30-3:45pm), the final exam is on Thursday December 13, 1-3pm.