Course Description

Instructor: Reza M. Rad
Office: ITE 374
Office Hours: Tue & Thu 9:30 –11:00AM
Wed 4:30-6:00 PM
Telephone: 455-8776
Email: reza2@umbc.edu
Web: www.csee.umbc.edu/~reza2

Teaching Assistants:
Niranjan Bhosarekar, Office Hours: Mon-Wed 2:30-3:45pm
Kuo-Hsin Tu, Office Hours: Tue-Thu 2:15-3:30 PM

Time and Place.
Tue & Thu 5:30pm – 6:45pm, ITE 233.
Tue & Thu 1:00pm – 2:15pm, ITE 233.

Textbooks.

- *Assembly Language for Intel-based Computers, Kip Irvine, Fifth Ed., prentice Hall 2007*
- *Logic and Computer Design Fundamentals, M. Morris Mano, Charles R. Kime, fourth Ed, Prentice Hall, 2008*

Supplementary Text:
- *Principles of Computer Architecture,* Murdocca and Heuring, Prentice-Hall, 2000.
- *Linux Assembly Language Programming,* Neveln, Prentice-Hall, 2000.

Prerequisites. You should have mastered the material covered in the following courses: CMSC 202 Computer Science II and CMSC 203 Discrete Structures. In particular, we will assume that you have had extensive programming experience in C/C++.

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 and computer organization. Under the heading of assembly language programming students will be introduced to the i386 instruction set, low-level programming and the memory model. 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). A secondary goal of this course is to prepare computer science majors for CMSC 411 Computer Architecture.

Grading. Your final grade will be based upon 5 homework assignments (20% total), 3 short programming assignments (12% total), 1 long programming assignment (8%), 2 circuit simulation exercises (10% total), four quizzes (8% total), a midterm exam (21%) and a final exam (21%). Your grade is given for work done *during* the semester; incomplete grades will only be given for medical illness or other such dire circumstances.

Lecture Policy. You are expected to attend all lectures. You are responsible for all material covered in the lecture as well as those in the assigned reading. However, this subject cannot be learned simply by listening to the lectures and reading the book. In order to master the material, you need to spend time outside the classroom on the programming assignments, simulation exercises and homework assignments.

Due Dates. There will be homework or exercises due on most weeks. Homeworks are due at the **beginning** of lecture. Exercises and projects turned in via online submissions are due 1 minute past 11:59pm of the due date. ***With one exception, late homework, exercises and programming assignments will not be accepted — this is to allow for timely grading and discussion of the solutions. The exception is that 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, exercises and projects must be completed by individual effort. 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.
Cases of academic dishonesty will be dealt with severely.

Exams. The exams will be closed-book and closed-notes. The date for the midterm exam is Tuesday, October 21st. The final exam will cover the material from the second part of the course. The date and time of the final exams will be posted on UMBC website.
Advising Note. This course is a replacement for CMSC 211 Assembly Language Programming and CMSC 311 Computer Organization which are no longer offered at UMBC. However, computer science majors who take this class must also take CMSC 411 Computer Architecture to satisfy the requirements of a BS degree in computer science. CMSC 313 by itself will not be sufficient for graduation — even if you've taken CMSC 211 or CMSC 311 previously.

CMSC 313, Computer Organization & Assembly Language Programming Fall 2008

Course Syllabus

We will follow two textbooks: *Assembly Language for Intel-based Computers and Logic and Computer Design Fundamentals. Few of the lectures are based on the supplementary text: Principles of Computer Architecture,* Murdocca and Heuring. The following schedule outlines the material to be covered during the semester and specifies the corresponding sections in each textbook.

Date	Topic	Text/Chapter		Assign	Due
		Irvine	**Mano**		
Thu 08/28	Introduction		Chap1(M&H)		
Tue 09/02	Data Representation & basic Logic operations	Chap 1			
Thu 09/04	Digital System overview and Data Representation		Chap1	Hw1	
Tue 09/09	Processor Architecture	Chap 2		Quiz1	
Thu 09/11	Processor Architecture	Chap 2		Hw2	Hw1
Tue 09/16	Linking, Compiling and Visual C++ Express tutorial	Chap 3		Proj1	
Thu 09/18	Assembly Lang. Fundamentals	Chap 3			Hw2
Tue 09/23	Data Transfer,	Chap4			Proj1
Thu 09/25	Conditions & Loops	Chap 6		Quiz2	
Tue 09/30	Integer Arithmetic	Chap 7			
Thu 10/02	MS-DOS Programming	Chap13		Proj2	
Tue 10/07	Strings & Arrays	Chap 9			
Thu 10/09	Structures & Macros	Chap 10		Proj3	Proj2
Tue 10/14	Procedures & Stack Frames,	Chap 5,8			
Thu 10/16	High Level Language Interface	Chap 12		Proj4	Proj3
Tue 10/21	**Midterm Exam**				
Thu 10/23	Introduction to Digital Logic		Chap 2		
Tue 10/28	Combinational Circuits		Chap 2	Hw3	Proj4
Thu 10/30	Combinational Circuits		Chap 2		
Tue 11/04	Combinational Design		Chap 3	Hw4	Hw3
Thu 11/06	Combinational Design		Chap 3	Quiz3	
Tue 11/11	Arithmetic Functions		Chap 4	Sim1	Hw4
Thu 11/13	Sequential Circuits		Chap 5		
Tue 11/18	Sequential Circuits/state machines		Chap 5		
Thu 11/20	Sequential Circuits/state machines		Chap 5	Sim2	Sim1
Tue 11/25	Sequential Circuits/state machines		Chap 5	Hw5, Quiz4	
Thu 11/27	**Thanksgiving**				
Tue 12/02	Registers & Counters		Chap7		Sim2
Thu 12/04	Memory Basics		Chap8		Hw5
Tue 12/09	Memory, Virtual Memory, Paging, Cache ..		Chap7(M&H)		

Policy on Programming Projects and Exercises

Critical programming skills cannot be learned by attending lecture. You should budget enough time to work on the programming assignments as well. Please consult the time table given on the syllabus and plan ahead. Programs are due by midnight (1 minute after 11:59pm) of the due date. Programs will be submitted using the `submit` system running on the GL machines. Late assignments will not be accepted (with the one exception noted in the course description). Programs will be graded on five criteria: correctness, design, style, documentation and efficiency. So, turning in a project that merely "works" is not sufficient to receive full credit.

For this course, programming projects must be developed using the MASM8 assembler for the Windows operating system running on an Intel Pentium CPU. All examples of the text are implemented in Visual C++ Express that provides an easy to use environment for coding, compiling and debugging. Hence, it is recommended that the course projects be developed in this environment.

Cheating

Read this section carefully! It describes what constitutes cheating for this course. If you have questions, ask the instructor. Ignorance will not be accepted as an excuse after the fact.

All programming assignments and circuit simulation exercises must be completed by your own individual effort. You should never have a copy of someone else's program either on paper or electronically under any circumstance. Also, you should never give a copy of your program or circuit, either on paper or electronically, to another student. This also means that you cannot work on the programming assignments or circuit simulation exercises together. Cases of academic dishonesty will be dealt with severely.

Egregious cases of cheating will be reported as a major infraction. In this case, you will not be allowed to drop the course. Also, a major infraction would appear as a permanent part of your student record and would be seen by potential employers when they ask for an official copy of your transcript.

We will be using special software to check for cheating. The software is quite sophisticated, has been tuned for assembly language programs and has surprised some students in the past. We will, of course, not release the details of the internal workings of this cheat-checking software, but you are forewarned that there is no difficulty in comparing every pair of submitted projects.

Finally, you are also warned that if your program is turned in by someone else, then, at a minimum, both you and the person who copied your program will receive a 0 for that assignment. This includes substantially similar programs. Furthermore, all parties concerned will have their prior programs checked for cheating. So, if you cheat on the last assignment, you can lose all the points from all of your assignments — even if you did all the work and just let other people copy from you.