CSCI/CMPE 2333 - Computer Organization and Assembly Language

Fall 2016

Instructor: David L. Egle Office: Engr 3.296
Phone: 665-3518 E-mail: david.egle@utrgv.edu
Website: faculty.utrgv.edu/david.egle/csci2333

Office Hours: MTWR 1:30-2:30, F 2-3
Other times by appointment

Text: Assembly Language for x86 Processors, 7th Ed., by Kip R. Irvine,
ISBN 978-0-13-376940-1 (Required)
Computer Organization and Architecture: Designing for Performance, 9th Ed.,
by William Stallings, ISBN 978-0-13-293633-0 (Recommended)

Objectives: This course is intended to provide the student with an introduction to computer organization and assembly language programming. Its purpose is to provide the student with a better understanding of the internal operation of the computer.

Prerequisites: This course part of the required sequence of introductory Computer Science and Computer Engineering courses. Students are expected to have successfully completed CSCI/CMPE 1370, or have the consent of the instructor. **(If you do not meet these requirements, you will be dropped from the course.**) This course must be successfully completed (with a grade of 'C' or better) to continue the course of studies in Computer Science.

Grading: 2 major exams, worth 40% of the course grade
Assigned homework, worth 10% of the course grade
Programming assignments, worth 30% of course grade
Final exam, worth 20% of the course grade

Drops: The last day to drop the course is November 17, 2016 (date subject to change). If a student has not turned in any homework or programming projects, has missed an exam, or is otherwise failing the course, that student will be dropped by the instructor by November 18, 2015 unless the student meets with me and convinces me that he or she will be able to pass the course.

Students considering dropping the class should be aware of the "3-peat rule" and the "6-drop" rule so they can recognize how dropped classes may affect their academic success. The 6-drop rule refers to Texas law that dictates that undergraduate students may not drop more than six courses during their undergraduate career. Courses dropped at other Texas public higher education institutions will count toward the six-course drop limit. The 3-peat rule refers to additional fees charged to students who take the same class for the third time.

Assignments: There will be about 5-6 programs assigned. These are expected to be organized and well-documented. The specific details for grading and documentation will be given at the time of the first program assignment.

Assignments will be graded on correctness, quality, and style. You MUST submit

ALL homework/programming projects, with no exceptions, in order to get overall credit for the assignments/programming projects. All homework must be turned in as a hard-copy and must be completely legible. Any portion which is not clearly and easily legible will receive a 0. All programming projects must be submitted using the tool provided by Blackboard. I will not accept programming projects through e-mail or by hard-copy. Also, all programming projects must compile or they will receive a 10% AT MOST.

SCHOLASTIC INTEGRITY: As members of a community dedicated to Honesty, Integrity and Respect, students are reminded that those who engage in scholastic dishonesty are subject to disciplinary penalties, including the possibility of failure in the course and expulsion from the University. Scholastic dishonesty includes but is not limited to: cheating, plagiarism, and collusion; submission for credit of any work or materials that are attributable in whole or in part to another person; taking an examination for another person; any act designed to give unfair advantage to a student; or the attempt to commit such acts. Since scholastic dishonesty harms the individual, all students and the integrity of the University, policies on scholastic dishonesty will be strictly enforced (Board of Regents Rules and Regulations and UTRGV Academic Integrity Guidelines). All scholastic dishonesty incidents will be reported to the Dean of Students.

Therefore, you are expected to create your own programs with no assistance from any source other than your notes, textbook, and your instructor. Copying, collusion, and other forms of plagiarism will not be tolerated. Programs which are not your own work will not earn any credit.

Late policy: All assignments should be turned in on their due date. Programs turned in late will be graded on the following basis:

no more than 1 week late:	10 point penalty
no more than 2 weeks late:	50 point penalty
more than 2 weeks late:	no credit given (grade is 0)

Make-up exams will not be given except by my prior consent. You must notify me within 24 hours after missing the exam so that I may determine the appropriateness of allowing a make-up exam. Examples of acceptable excuses would be the death of an immediate family member, or an illness requiring physician's attention. You must take all exams in order to pass the course; missing any one exam will result in an 'F' as your course grade.

NOTE: If you have a documented disability (physical, psychological, learning, or other disability which affects your academic performance) and would like to receive academic accommodations, please inform your instructor and contact Student Accessibility Services to schedule an appointment to initiate services. It is recommended that you schedule an appointment with Student Accessibility Services before classes start. However, accommodations can be provided at any time. Student Accessibility Services is located in 108 University Center and can be contacted by phone at (956) 665-7005 (Voice), (956) 665-3840 (Fax), or via email at accessibility@utrgv.edu.

The following is a general outline for the course and may be revised as the semester progresses. In this, ST is Stallings book, and IR is Irvines book.

Week	Topic	Chapter
1-2	Computer Evolution; Top-level view of a computer	ST: 1-3
3	Computer Memory System Overview	ST: 4
4	Internal & External Memory	ST: 5-6
5	Computer Numbers and Arithmetic	ST: 9
6	**Exam 1**	
7	Assembly language fundamentals; LAB	IR: 3
8	Data transfer; Arithmetic; Addressing	IR: 4
9	Procedures and parameter passing	IR: 5
10	**Exam 2**	
11	Logic and decision instructions	IR: 6
12	Integer arithmetic	IR: 7
13	Advanced procedures	IR: 8
14-15	Misc topics as time permits	

Final Exam – Dec 15, 1:15-3pm

Mandatory Course Evaluation Period: Students are required to complete an ONLINE evaluation of this course, accessed through your UTRGV account (http://my.utrgv.edu); you will be contacted through email with further instructions. Online evaluations will be available Nov. 18 – Dec. 8, 2016. Students who complete their evaluations will have priority access to their grades.

SEXUAL HARASSMENT, DISCRIMINATION, and VIOLENCE: In accordance with UT System regulations, your instructor is a "responsible employee" for reporting purposes under Title IX regulations and so must report any instance, occurring during a student's time in college, of sexual assault, stalking, dating violence, domestic violence, or sexual harassment about which she/he becomes aware during this course through writing, discussion, or personal disclosure. More information can be found at www.utrgv.edu/equity, including confidential resources available on campus. The faculty and staff of UTRGV actively strive to provide a learning, working, and living environment that promotes personal integrity, civility, and mutual respect in an environment free from sexual misconduct and discrimination.

Learning outcomes:

At the end of this course, the student should be able to

1. Describe the progression of computer architecture from vacuum tubes to VLSI.
2. Demonstrate an understanding of the basic building blocks and their role in the historical development of computer architecture.
3. Design a simple circuit using the fundamental building blocks.
4. Explain how interrupts are used to implement I/O control and data transfers.
5. Identify various types of buses in a computer system.
6. Explain the reasons for using different formats to represent numerical data.
7. Explain how negative integers are stored in sign-magnitude and twos-complement representation.
8. Convert numerical data from one format to another.
9. Discuss how fixed-length number representations affect accuracy and precision.
10. Describe the internal representation of nonnumeric data.
11. Describe the internal representation of characters, strings, records, and arrays.
12. Explain the organization of the classical von Neumann machine and its major functional units.
13. Explain how an instruction is executed in a classical von Neumann machine.
14. Write assembly language programs that use basic computation and simple I/o, standard conditional structures, basic iterative structures, and the definition of functions.
15. Demonstrate how fundamental high-level programming constructs are implemented at the machine-language level.
16. Explain how subroutine calls are handled at the assembly level.
17. Explain the basic concepts of interrupts and I/O operations.
18. Choose appropriate conditional and iteration constructs for a given programming task.
19. Describe the mechanics of parameter passing.

ABET Student Learning Outcomes:

(a) An ability to apply knowledge of computing and mathematics appropriate to the discipline.

(b) An ability to analyze a problem, and identify and define the computing requirements appropriate to its solution.

(c) An ability to design, implement, and evaluate a computer-based system, process, component, or program to meet desired needs.

(i) An ability to use current techniques, skills, and tools necessary for computing practice.