CIS 123 - Assembly Language and Computer Architecture – SPRING 2011

INSTRUCTOR: Ron Mummaw (**moo**-maw) OFFICE: BE 210

PHONE: 722-6379 E-MAIL: rmummaw@avc.edu

WEB SITE: avconline.avc.edu/rmummaw/cis123

OFFICE HOURS: M-R -- 7:40-8:30am; 10:35-11:00am

TEXTBOOK: Assembly Language for Intel-based Computers/Sixth Ed., Kip Irvine ISBN-10: 0-13-602212-X

REQUIRED: **Every class, you must have** 1) a flash drive, 2) pen and pencil, 3) paper, 4) textbook

ADVISORY: Completion of CIS 111, and Eligibility for ENGL 099, READ 099 and MATH 102.

DESCRIPTION: This course introduces assembly language programming and computer architecture to enable students to understand how programs are actually executed at the machine level. Students will use Intel or compatible processor personal computers for the detailed study of the Intel IA-32 processor instruction set and architecture and to develop programs using a macro assembler. Both 32-bit Windows console programming and 16-bit real-mode programming are covered. Topics include machine/assembly level programming, instruction formats, internal data representation, addressing modes, procedure call and return mechanisms, and how high-level language constructs are implemented at the machine level, basic microcomputer organization, instruction execution cycle, memory segmentation and paging, and details of programming the processor in both protect-mode and in real-mode. **BEFORE ENROLLING** students should have basic computer experience and be able to save and retrieve files, run applications, print documents, and have sufficient aptitude with mathematics to solve simple algebraic equations and to appreciate the use of mathematical notation and formalism. The student should also have completed with a passing grade, one semester of a programming class.

ASSIGNMENTS: During the course of the semester, you will be assigned one assignment per class. Obviously copied work will be graded only once and the grade divided among those students who turned them in. You will be expected to turn in assignments at the beginning of the next class. They will, however, be accepted up to 6:00pm two weeks later. As assignment not turned in after two weeks will be given **NO** credit, ***REGARDLESS*** of the reason. No time will be given at the beginning of class for printing.

**IMPORTANT STUFF TO REMEMBER IF YOU WANT 10 POINTS ON AN ASSIGNMENT!!!**

1. The first thing I want to see on the top of your paper is your name on one line all by itself. On the second line, I want to see the chapter number.

2. Assignments must be turned in at the BEGINNING of class on ***or before*** the deadline to receive credit. NO CREDIT WILL BE GIVEN TO LATE WORK (no exceptions). You will not be given time in class to print out any part of an assignment on the day of the deadline.

3. A paper containing only your name and the assignment will receive one point.

EXAMS: Three exams will be given. Two exams during the semester and one final exam. The first exam will be broken into 3 parts; 1/chapter. All three are equal credit. You will not be allowed to make up a missed exam unless I receive prior notice that you are going to be absent. You will be expected to be prepared to take the exam on your return.

GRADING: class participation 40 points

assignments 11 @ 10 points each 110 points

tests 5 @ 30 points each 150 points

total 300 points

270 - 300 A

240 - 269 B

etc.

Schedule (this is **TENTATIVE**)

|  |  |
| --- | --- |
| 02.08 | Intro |
| 02.15 | Chapter 1 |
| 02.22 | Chapter 2 – TEST CHAPTER 1 |
| 02.29 | Chapter 3 – TEST CHAPTER 2 |
| 03.07 | Chapter 4.1-4.3 |
| 03.14 | TEST CHAPTER 3 |
| 03.21 | Chapter 4.4-4.5 |
| 03.28 | Lab |
| 04.04 | SPRING BREAK |
| 04.11 | Chapter 5 |
| 04.18 | Chapter 6 |
| 04.25 | Chapter 7 |
| 05.02 | TEST CHAPTERS 4-6 |
| 05.09 | Chapter 8 |
| 05.16 | Chapter 9 |
| 05.23 | Lab |
| 05.30 | Final Exam Chapters 7-9 |

ASSIGNMENTS:

You will be expected to turn in assignments at the beginning the next class after they are assigned, but will be accepted no later than 6:00pm two weeks after they are assigned. As assignment not turned in after two weeks will be given NO credit, REGARDLESS of the reason. No time will be given at the beginning of class for printing.

LAB RULES:

1. Food and drinks are not allowed on the third floor of this building with the exception of bottled water. There will be no exceptions to this rule.

2. Computers must be turned off during the lecture unless otherwise specified.

ATTENDANCE: More than two(2) absences or 4 tardies will result in you being dropped from the class. I may make exceptions to this if I am given advance notice and there is a legitimate reason. It is recommended that you become friends with another student in class and exchange email addresses or phone numbers so that you can contact them in case you do have to miss class. Be here and BE ON TIME.

COURSE OBJECTIVES:

Upon completion of course, the successful student will be able to:

1. Plan, code, run, debug, and document programs written in assembly language.

2. Describe the internal representation of numerical and nonnumeric data.

3. Demonstrate how fundamental high-level programming constructs are implemented at the machine-language level.

4. Select the best mode of addressing when accessing different types of data from memory.

5. Choose the appropriate call-return mechanism when interfacing to high-level language code.

6. Explain how interrupts are used to implement I/O control, data transfers, and "calls" to system procedures.

7. Describe the components of a microcomputer system.

8. Describe the processor's fetch, decode, and execute cycle and indicate how an interrupt is handled.

9. Explain the concept of virtual memory and how it is realized in hardware and software.

10. Describe the reason for and use of cache memory and instruction pipelining.

COURSE OUTLINE:

|  |  |
| --- | --- |
| I. Basic Concepts  a. Components of a microcomputer  b. Virtual Machine Concept  c. Boolean operations  d. Truth tables  e. Binary integer arithmetic  II. Assembly language programming  a. Assembling, linking, and executing  b. Instructions vs. Directives  c. Instruction format  d. Defining data and structures  e. Data addressing modes  f. Using the processor's instructions  1. Data transfers  2. Integer arithmetic  3. Branching and Looping  4. Conditional operations  5. "String" instructions  6. Bit manipulation instructions  g. Procedures  1. Defining procedures  2. Call-return mechanisms  3. Linking to a library  4. Stack operations  h. High-Level Language Interface  i. Directives that control assembling  j. Defining Segments  k. Defining Macros  l. Debugging techniques | III. Microprocessor Architecture  a. Instruction execution cycle  b. Functional organization  1. data buses  2. control unit and clock  3. arithmetic/logical unit  4. instruction pipelining  c. Memory organization  1. Segmentation  2. Paging  d. Protected mode vs. Real mode  e. Program execution registers  f. Instruction representation  g. Complex vs. Reduced instruction set  h. Internal data representation  i. Input, Output, and Interrupts  j. Floating-point processor  IV. Operating System Interface  a. 32-bit Windows console programming  b. 16-bit DOS programming  1. Interrupt handling  2. BIOS hardware interface |

Antelope Valley College prohibits discrimination and harassment based on sex, gender, race, color, religion, national origin or ancestry, age, disability, marital status, sexual orientation, cancer-related medical condition or genetic predisposition. Upon request, we will consider reasonable accommodation to permit individuals with protected disabilities to (a) complete the employment or admission process, (b) perform essential job functions, (c) enjoy benefits and privileges of similarly-situated individuals without disabilities, and (d) participate in instruction, programs, services, activities or events.