Las Positas College
3000 Campus Hill Drive
Livermore, CA 94551-7650
(925) 424-1000
(925) 443-0742 (Fax)

Course Outline for CS 21

COMPUTER ORGANIZATION AND ASSEMBLY LANGUAGE PROGRAMMING

Effective: Fall 2005

I. CATALOG DESCRIPTION:
CS 21 — COMPUTER ORGANIZATION AND ASSEMBLY LANGUAGE PROGRAMMING — 4.00 units

Basics of machine architecture, machine language, assembly language, operating system and higher level language interface. Data representation, instruction representation and execution, addressing techniques and use of macros. Space and time efficiency issues. Input/output including video modes. Procedures including parameter passing and linkage to higher level languages.

3.00 Units Lecture 1.00 Units Lab

Prerequisite
CS 1 - Computing Fundamentals I
with a minimum grade of C

Grading Methods:
Letter or P/NP

Discipline:

	MIN
Lecture Hours:	54.00
Lab Hours:	54.00
Total Hours:	108.00

II. NUMBER OF TIMES COURSE MAY BE TAKEN FOR CREDIT: 1

III. PREREQUISITE AND/OR ADVISORY SKILLS:
Before entering the course a student should be able to:

A. CS1

IV. MEASURABLE OBJECTIVES:
Upon completion of this course, the student should be able to:

A. explain computer architecture and operating system interface;
B. demonstrate machine addressing techniques;
C. demonstrate how data is represented in the machine: integers, characters, strings, floating point numbers (in IEEE representation), arrays;
D. handle integer arithmetic and string manipulations efficiently;
E. program selection and repetition constructs, macros and procedures in assembly language including parameter passing and linkage to both external assembly language modules and higher level language modules;
F. program basic keyboard input and text screen output using operating system interrupts and library procedures;
G. perform output in graphics screen modes;
H. explain how floating point arithmetic is performed using the co-processor;
I. program elementary file input/output in assembly language.

V. CONTENT:
A. Computer architecture
    1. The central processor registers
    2. Communication with memory
    3. Instruction format and execution cycle
B. Addressing techniques
    1. Absolute vs relative addressing
    2. Indexed addressing
    3. Indirect addressing
C. Machine and assembler representation of data
    1. Base arithmetic and base conversion
    2. Integers of byte and word size and larger
    3. Characters, character strings and arrays
    4. Floating point numbers in IEEE format
D. Assembly language programming fundamentals

1. Assembly format including labels, operation codes, operands and remarks
2. The assembly process and production/meaning of source, object, listing, map and executable files
3. Pseudo ops
4. Machine format of instructions
5. Use of the registers and moving data to/from memory
6. Integer arithmetic and logical machine instructions
7. Basic input from keyboard and output to screen of integers, characters and strings using DOS interrupts and library routines and prewritten macros
8. Test and compare instructions and use in selection constructs
9. Branch, jump and loop instructions and use in repetition constructs
10. Index register modified instructions and use with arrays
11. Procedures which do not explicitly pass parameters

E. Advanced assembly language topics
1. Procedures including parameter passing using the stack
2. and stack/base pointer registers
3. Writing macros
4. Input and output to files
5. Video modes
6. Floating arithmetic using the co-processor
7. Advanced use of segments and segment registers
8. How an assembler is written

F. Program development
1. Error detection using on line debugging programs
2. Automating the assembly, link, debug cycle
3. Program linkage with externally assembled procedures
4. Program linkage with higher level languages

VI. METHODS OF INSTRUCTION:

A. **Lecture** -
B. **Demonstration** -
C. **Projects** - Optional: Programming projects completed in teams
D. **Lab** - Lab Programming Assignments
E. **Discussion** -

VII. TYPICAL ASSIGNMENTS:

A. Write an assembly program to input two integers. Compute and display the sum, difference, and average of the two numbers. B. Write an assembly program to input a string of characters and display them backwards. C. Write an assembly program to input a set of numbers from a data file and display the average of the numbers. Be sure to handle the case where there are no numbers in the file. This should be done for a file of integers and for a file of real numbers.

VIII. EVALUATION:

A. **Methods**

B. **Frequency**

1. Frequency
   a. At least two in-class midterm examinations, or one in-class midterm examination and several quizzes
   b. Additional midterm examination(s) and/or quizzes
   c. In-class comprehensive final examination
   d. Programming assignments to cover each topic within the course content (contents can be combined). At least two of the programming assignments must be in-class, or covered by in-class examinations. It is suggested that there be at least one programming assignment each week.
2. Types of Exam Questions
   a. Write an assembly program that incorporates inputs and determines smaller of two integer values.
   b. Write a program that inputs two integers from a file and outputs their sum and difference.
   c. Convert the decimal number 231 to a hexadecimal and octal.
3. Evaluation Percentage Recommendations
   a. In-class grading percentage: 65% (In-class percentage may be increased, but not decreased.) For example:
      1. In-class Programming Assignments: 10% (May be included in the midterm and final exams.)
      2. In-class Midterm(s) examination: 20%
      3. In-class Final examination: 35%
   b. Other: 35% (e.g. programming assignments, chapter quizzes, written assignments, student presentations, etc…)

Note: It is suggested that programming assignments be of substantial difficulty to verify mastery of the subject area.

IX. TYPICAL TEXTS:

1. William B. Jones *Assembly Language for the IBM PC Family.*, Scott/Jones, 2002.
2. Kip R. Irvine *Assembly Language for Intel-based Computers:.* 4th ed., Prentice Hall, 2003.

X. OTHER MATERIALS REQUIRED OF STUDENTS:

A. Computer Access Diskette