GAVILAN COLLEGE
5055 Santa Teresa Blvd
Gilroy, CA 95023

Course Outline

COURSE: CSIS 12 DIVISION: 50 ALSO LISTED AS:

TERM EFFECTIVE: Fall 2016 CURRICULUM APPROVAL DATE: 11/23/2015

SHORT TITLE: ASSEMBLY LANGUAGE

LONG TITLE: Assembly Language Programming

Units	Number of Weeks	Type	Contact Hours/Week	Total Contact Hours
3	18	Lecture:	3	54
		Lab:	0	0
		Other:	0	0
		Total:	3	54

COURSE DESCRIPTION:

Fundamentals of assembly language programming concepts and techniques. Topics include internal representation of data, arithmetic operations, logic statements, and general assembly language commands. Introduce low level language architecture including assemblers, linkage editors, and loaders. This course has the option of a letter grade or pass/no pass. COREQUISITE: CSIS 12L Assembly Language Programming Lab ADVISORY: CSIS 45 (C++ Programming) or programming experience. Math 233 (Intermediate Algebra)

PREREQUISITES:

COREQUISITES:
CSIS 12L

CREDIT STATUS: D - Credit - Degree Applicable

GRADING MODES
L - Standard Letter Grade
P - Pass/No Pass

REPEATABILITY: N - Course may not be repeated

SCHEDULE TYPES:
02 - Lecture and/or discussion
05 - Hybrid
72 - Dist. Ed Internet Delayed

STUDENT LEARNING OUTCOMES:

1. Create, modify, execute, debug, and print a simple assembly language program..
Measure: Homework, exercises, quizzes.
PLO: 2
ILO: 7, 3, 2
GE-LO:
Year assessed or anticipated year of assessment: 2016

2. Create, modify, execute, debug, and print an assembly language program that uses three types of loops.
Measure: Homework, exercises, quizzes.
PLO: 2, 4
ILO: 3, 7, 2
GE-LO:
Year assessed or anticipated year of assessment: 2016

3. Create, modify, execute, debug, and print an assembly language program that uses accumulators, registers and hexadecimal numbers.
Measure: Homework lab exercises, projects
PLO: 2, 4
ILO: 3, 7, 2
GE-LO:
Year assessed or anticipated year of assessment: 2016

4. Create, modify, execute, debug, and print an assembly language program that uses decision and jump statements.
Measure: Exercises, homework, quizzes
PLO: 2, 3, 4
ILO: 3, 7, 2
GE-LO:
Year assessed or anticipated year of assessment:

5. Create, modify, execute, debug, and print an assembly language program that uses five different arithmetic operations and four arithmetic functions.
Measure: Homework, quizzes, projects.
PLO: 2, 4, 3
ILO: 7, 3, 2
GE-LO:
Year assessed or anticipated year of assessment:

PROGRAM LEARNING OUTCOMES:

1) Student will code, debug, document, test, and run complex C++ programs.

2) Student will write programs in at least three different programming languages, and compare and contrast the philosophies and comparative advantages of each these languages.

3) Students will demonstrate professional conduct by meeting project deadlines, and participating in self-managed teams.

4) Student will create algorithms to solve programming problems, and implement those algorithms.

CONTENT, STUDENT PERFORMANCE OBJECTIVES, OUT-OF-CLASS ASSIGNMENTS

Curriculum Approval Date: 11/23/2015

WEEK HOURS CONTENT

1-2 L Lecture

6 What is assembly language? Who uses it? Why is it necessary?
Variety of assembly languages for different CPU platforms - 8 bit, 15 bit and 32 bit systems.
Assembly language vs. machine code
Historical background
Number system
How to use the number system to represent data formats.
Homework:
Read chapter in the textbook related to introduction to assembly language and number system.
Practice number conversions: binary to hex; hex to binary.
Do exercise related to these topics.
Read chapter in the textbook related to register types.

3-4 L Lecture:

6 How accumulator, base index, counter and data registers are modeled in an 8 bit, 16 bit and 32 bit machine.
What is real mode memory addressing?
What are segments and offsets?
Introduction to protected mode memory addressing.
What is memory paging?
Homework:
Selected questions and problems from the end of the chapter.
Read chapter in the textbook related to addressing.
Chapter test.

5-7 L Lecture:

9 What are data addressing modes?
Opcode: MOV.
What are register addressing, immediate addressing, direct addressing, register indirect addressing, base-plus-index address, register relative addressing, base relative-plus-index addressing and scaled index addressing?
How these addressing modes are implemented?
How they are used in assembly language programming?
What are the stack memory addressing modes?
Homework:
Selected questions and problems from the end of the chapter.
Read chapter in the textbook related to other move

instructions.
8-10 L Lecture:
9 Data Move instructions.
Opcode: MOVSX, MOVZX, PUSH, POP, BSWAP, XCHG, XLAT,
IN, OUT, LEA, LDS, LES, LFS, LGS, LSS, HAFH, SAHF,
String instructions.
Opcode: MOVS, LODS, STORS, INS, OUTS
Mid term Exam
Homework:
Selected questions and problems from the end of the
chapter.
Read chapter in the textbook related to ALU, arithmetic
and logic instructions sets
11-12 L Lecture:
6 What are ALU, arithmetic and logic instructions?
Arithmetic instruction: add,
subtract,
multiplication,
division,
negation,
comparison,
increment,
and decrement.
Homework:
Selected questions and problems from the end of the
chapter.
Read chapter in the textbook related to logic
instruction sets.
13-14 L Lecture:
6 Logic instructions: AND,
OR,
Exclusive-OR,
NOT,
shifts,
rotates,
and logical compare (TEST).
Homework:
Selected questions and problems from the end of the
chapter.
Read chapter in the textbook related to jump
instruction sets.
15-16 L Lecture:
6 Jump instruction set:
JA, JAE, JB, JBE, JC, JG, JGE, JL, JLE, JNC, JNE, JNO,
JNS, JNP, JO, JP, JS, JCXZ, JECXZ.
Homework:
Selected questions and problems from the end of the
chapter.

Read chapter in the textbook related to macros.
17-18 L Lecture:
6 Use the MASM assembler and linker program.
Use EXTRN and PUBLIC
Set up library files
Write and use MACRO and ENDM to develop macro
sequences.
Final Exam
Homework:
Selected questions and problems from the end of the
chapter.
Read chapter in the textbook related to programming
with C++
STUDENT PERFORMANCE OBJECTIVES:
Weeks 1-2
Students should have a thorough understanding of the number system and how it's used in the computer system.
Students should learn the relationship between, high-level language and assembly language.
Students should learn how memory maps are used in a computer system.
Overview of the DOS operating system.
Weeks 3-4
Students should understand how to model accumulator, base index register, counter and data register in 8 bit, 16 bit and 32 bit environment. How to use the registers to accomplish memory addressing? Students should understand the difference between real mode memory addressing, protected mode memory addressing and how to implement them in a computer system.
Weeks 5-7
Student should learn the operation of each data-addressing mode.
Students should be able to explain the operation of each program memory-addressing mode and select the appropriate one to accomplish certain tasks. Describe the sequence of events that place data onto the stack or remove data from the stack.
Weeks 8-10
Students should be able to explain the operation of each data movement instruction with applicable addressing modes.
Understand the purposes of the assembly language pseudo-operations and key words.
Select the appropriate assembly language instruction to accomplish a specific date movement task.
Weeks 11-12
Students should be able to use arithmetic and logic instructions to accomplish simple binary, BCD, and ASII arithmetic.
Weeks 13-14
Use AND, OR and Exclusive-AND, to accomplish binary bit manipulation.
Use the shift and rotate instruction.

Weeks 15-16
Use both conditional and unconditional jump instructions to control
flow of a program.
Use the relational assembly language statements: IF, .REPEAT, .WHILE
Use the call and return instructions to include procedures in
program structure.
Weeks 17-18
Write assembler code using macros
Able to link and include libraries
ASSIGNMENTS:
Included in the content section of the course outline.

METHODS OF INSTRUCTION:
Lecture, computer demonstration, projects, lab exercises.

METHODS OF EVALUATION:
The types of writing assignments required:
Written homework
Reading reports
Lab reports
The problem-solving assignments required:
Homework problems
Field work
Exams
The types of skill demonstrations required:
Class performance
Performance exams
The types of objective examinations used in the course:
Multiple choice
True/false
Matching items
Completion
Other category:
None
The basis for assigning students grades in the course:
Writing assignments: 10% - 40%
Problem-solving demonstrations: 30% - 50%
Skill demonstrations: 10% - 50%
Objective examinations: 5% - 20%
Other methods of evaluation: 0% - 0%

REPRESENTATIVE TEXTBOOKS:
Required:
Irvine, Kip. Assembly Language for x86 Processors (7th Edition). Pearson, 2014. Or other appropriate college level text.
Reading level of text, Grade: 12+ Verified by: ev

ARTICULATION and CERTIFICATE INFORMATION

Associate Degree:
CSU GE:
IGETC:
CSU TRANSFER:
Transferable CSU, effective 200770
UC TRANSFER:
Transferable UC, effective 200770

SUPPLEMENTAL DATA:
Basic Skills: N
Classification: Y
Noncredit Category: Y
Cooperative Education:
Program Status: 1 Program Applicable
Special Class Status: N
CAN:
CAN Sequence:
CSU Crosswalk Course Department: CSIS
CSU Crosswalk Course Number: 12
Prior to College Level: Y
Non Credit Enhanced Funding: N
Funding Agency Code: Y
In-Service: N
Occupational Course: C
Maximum Hours:
Minimum Hours:
Course Control Number: CCC000381891
Sports/Physical Education Course: N
Taxonomy of Program: 070710