Course Syllabus

Basic Information:

COLLAGE	
Collage Name	Collage of Engineering
Department	Computer Engineering
Semester	١st Semester
Year	2019/2020
COURSE	
Course Name	Computer Organization & Assembly Language
Course Code	ECOM 4412
Credits	4
Pre-requisite	ECOM 3421 Computer Architecture
TEACHER	
Teacher Name	Ruba A. Salamah
Office	L506
Phone int.	2819
Email	Rsalamah@iugaza.edu
Office Hours	Saturday: 11-12 Monday: 11-12 Tuesday 10:11

LECTURE	Building	Day	Start	End	Room Number
	L	Sunday Tuesday	8:00	9:30	L413

TEXTBOOK

"Assembly Language for x86 Processors", by Kip R. Irvine (7th Edition), Prentice-Hall, 2015.

REFERENACES

"Computer Organization and Embedded Systems", by C. Hamacher, Z. Vranesic, S. Zaky, and N. Manjikian, 6th ed., McGraw-Hill, 2012.

"Computer Organization And Design, the hardware / software interface", by David A. Patterson, and John L. Hennessy, 5th ed. 2014

Course Description:

Concepts of assembly language and the machine representation of instructions and data of a modern digital computer are presented. Students will have the opportunity to study machine addressing, stack operations, subroutines, and programmed and interrupt driven I/O. Students will utilize the Intel x86 instruction sets and will perform programming exercises using its assembly language.

Course Objectives:

1. Know basic principles of computer architecture as applied to x86 processors
2. Learn How x86 processors manage memory, using protected mode and virtual mode
3. Understand how high-level language compilers (such as C++) translate statements from their language into assembly language and native machine code
4. understand how high-level languages implement arithmetic expressions, loops, and logical structures at the machine level
5. Identify Data representation, including signed and unsigned integers, real numbers, and character data
6. Learn how to debug programs at the machine level. The need for this skill is vital when you work in languages such as C and C++, which generate native machine code
7. Learn how application programs communicate with the computer’s operating system via interrupt handlers and system calls
8. Learn how to create assembly language application programs

Course Contents:

Schedule Table

Week	Content	Notes
١	**Basic Concepts:** Applications of assembly language, basic concepts, machine language, and data representation.	
٢	**x86 Processor Architecture:** Basic microcomputer design, instruction execution cycle, x86 processor architecture, Intel64 architecture, x86 memory management, components of a microcomputer, and the input–output system.	
٣	**Assembly Language Fundamentals:** Introduction to assembly language, linking and debugging, and defining constants and variables.	
٤		
٥	**Data Transfers, Addressing, and Arithmetic:** Data transfer and arithmetic instructions, assemble-link-execute cycle, operators, directives, expressions, JMP and LOOP instructions, and indirect addressing.	
٦		

٧	**Procedures:** Linking to an external library, description of the book's link library, stack operations, defining and using procedures, flowcharts, and top-down structured design.	
٨		
٩	**Conditional Processing:** Boolean and comparison instructions, conditional jumps and loops, high-level logic structures, and finite-state machines.	
١٠		
١١	**Integer Arithmetic:** Shift and rotate instructions with useful applications, multiplication and division, extended addition and subtraction, and ASCII and packed decimal arithmetic.	
١٢	**Advanced Procedures:** Stack parameters, local variables, advanced PROC and INVOKE directives, and recursion.	
١٣	**Structures and Macros:** Structures, macros, conditional assembly directives, and defining repeat blocks.	
١٤	**MS-Windows Programming:** Protected mode memory management concepts, using the Microsoft-Windows API to display text and colors, and dynamic memory allocation.	

Outcomes:

A. Knowledge

1. Students will demonstrate knowledge of basic Computer Organization: design logic; digital diagrams, and basic circuits and gates, and the link between Boolean functions, circuits, processor and machine code
2. Name and define the elements of an assembly language program.
3. Describe and express in an assembly language program, data structures used to solve common engineering, mathematical, and business-oriented problems.

B. Cognitive Skills

1. write, test and debug programs in x86 assembly language using assembler, debugger and emulation software
2. Employ assembly language directives and operators to effectively express code and data in an assembly language program.
3. Use current software engineering techniques to design and implement programs.
4. Employ structured techniques to construct branching and looping in assembly code.
5. Diagram and analyze the use of the hardware stack when designing procedures.

C. Interpersonal skills and responsibilities

1. Relate x86 assembly language with other processor assembly languages and high-level languages.
2. Able to understand the organization of different processors other than the x86 CPU.
3. Earn teamwork skills and display potential leadership qualities

D. Analysis and communication

1. Evaluate the efficiency of an assembly language program.
2. skills and motivation for independent learning and engagement in lifelong learning and research

Teaching Strategies:

1- Power point presentation Lecture

2- Problem Solving

3- Discussion with small groups

4- Learning by doing

5- Scientific Research strategy

Course activities:

1. Lab work
2. Research
3. Presentation
4. Programming problems

Assesment:

1- MidTerm Exam 20%

2- Quizes 10%

3- Research 10%

4- Lab assignments 20%

5- Final Exam 40%

Technology Used:

You need a computer that runs a 32-bit or 64-bit version of Microsoft Windows, along with one of the recent versions of Microsoft Visual Studio, as well as MASM assembler