CS-201 COMPUTER ORGANIZATION AND ASSEMBLY LANGUAGE

Pre-requisite: CS-101 with a grade of C or better and MA-441  Hours: 3 Class Hours, 1 Lab Hour, 1 Recitation Hour, 4 Credits

Course Description: Principles of computer design and implementation. Instruction set architecture and register transfer level execution, storage formats, binary data encoding, bus structures and assembly language programming.

Curricula for which the course is required/recommended: A.S. Degree Programs in Liberal Arts and Sciences (Mathematics and Science), Electrical and Computer Engineering Technology.

General Education Objectives: Use analytical reasoning skills to identify issues or problems and evaluate evidence in order to make informed decisions; reason quantitatively and mathematically as required in their fields of interest and in everyday life; integrate knowledge and skills in their program of study; use information management and technology skills effectively for academic research and lifelong learning.

Course Objectives/ Expected Student Learning Outcomes: Understanding of basic computer architecture and the skill of programming at the assembly language level.


Methods by which student learning will be evaluated: The general guidelines for assessing grades are as follows:
- Examinations, Assignments : 60%
- Laboratory assignments : 15%
- Final Examination: 25%

Accommodations for Students with Disabilities: As stated in the current QCC catalog, any student who needs specific accommodations based upon the impact of a disability should register with the office of Services for Students with Disabilities (SSD) to be eligible for accommodations, which are determined on an individual basis. The SSD office is located in the Science Building, room S132, phone (718-631-6257). Students should also contact the instructor privately to discuss their specific needs.

Academic Integrity: Academic honesty is taken extremely seriously and is expected of all students. All assignments must be the original work of the student (and partners or group, if applicable). All questions or concerns regarding ethical conduct should be brought to the course instructor. “It is the official policy of the College that all acts or attempted acts that are violations 2 of academic integrity be reported to the Office of Student Affairs (OSA). At the faculty member’s discretion and with the concurrence of the student or students involved, some cases, though reported to the OSA, may be resolved within the confines of the course and department. The instructor has the authority to adjust the offender’s grades as deemed appropriate, including assigning an F to the assignment or exercise or, in more serious cases, an F to the student for the entire course.” (Taken from the QCC Academic Integrity Policy, 2/14/2005.)
1 Computer Implementation and Design
   1. Organization and Architecture
   2. Intel x86
   3. Memory System
   4. Interconnection System

2 Data Representation and Number Systems
   1. Decimal System
   2. Positional Number Systems
   3. Binary System
   4. Hexadecimal System
   5. Number System Conversion

3 Computer Arithmetic
   1. Arithmetic and Logic Unit
   2. Integer Representation
   3. Integer Arithmetic
   4. Floating Point Representation
   5. Floating Point Arithmetic
   6. Hexadecimal Addition & Subtraction
   7. Integer Multiplication

4 Digital Logic
   1. Boolean Algebra
   2. Logical Gates
   3. K-maps
   4. Combinatorial Circuits (MUX)
   5. Sequential Circuits (SR-latch FF)

5 Instruction Set Architecture
   1. Machine Instruction Characteristics
   2. Types of Operands
   3. Intel x86 Data types
   4. Intel x86 Operations

7 Basic Assembly Fundamentals
   1. Basic Assembly Language Concepts
   2. Adding and Subtracting Integers
   3. Linking and Running Programs
   4. Defining Data
   5. Symbolic Constants
   6. Multiplying and Dividing Integers

8 Data Transfers, Addressing & Arithmetic
   1. Data Transfer Instructions
   2. Addition and Subtraction
   3. Data Related Operators and Directives
   4. Indirect Addressing
   5. Jump and Loop Instructions

9 Assembly Procedures
   1. Stack Operations
   2. Defining and using Procedures
   3. Parameters and Special Directives

10 Conditional Processing
    1. Conditional Branching
    2. Boolean and Comparison Instructions
    3. Conditional Jumps
    4. Conditional Loops
    5. Conditional Structures
    6. Conditional Control Directives

11 Strings and Arrays
    1. String Instructions
    2. String Procedures
    3. Two-Dimensional Arrays

12 Structures and Macros
    1. Structures
    2. Macros

13 Advanced Topics
    1. Inline Assembly Code
    2. Floating Point Processing
    3. Dynamic Memory Allocation

Fall 2017
[RM CS 201]