Virginia Western Community College
CSC 205
Computer Organization

Prerequisites
CSC221

Course Description
Examines the hierarchical structure of computer architecture. Focuses on multi-level machine organization. Uses assembler programming to complete simple projects. Includes processors, instruction, execution, addressing techniques, data representation and digital logic.

Semester Credits: 3 Lecture Hours: 3 Lab/Clinical/Internship Hours: 0

Required Materials

Textbook:
None

Other Required Materials:
Computer with internet access sufficient to perform all coursework including watching videos, completing and submitting assignments online

Course Outcomes

At the completion of this course, the student should be able to:

- Machine level data representations
  - Describe how numbers, text, analog, and other information are stored in a digital form
  - Interpret computer data representation of unsigned integer, signed integer (in 2's complement form) and floating-point values in the IEEE-754 formats
  - Explain the impact due to the limitations of data representations such as rounding effects and how their propagation affects the accuracy of chained calculations, overflow errors, and mapping of continuous information to discrete representation
- CPU and Instruction set architectures
  - Differentiate various instruction set architectures
  - Correlate instruction set architecture, microarchitecture, and system architecture
  - Explain the concept of an instruction set architecture, ISA, and the nature of a machine-level instruction in terms of its functionality and use of resources (registers and memory).
  - Define the various classes of instruction: data movement, arithmetic, logical, and flow control
  - Describe the low-level operation and architecture of a typical CPU, including components, data pathways, and control signals
  - Explore Instruction Set Architecture (ISA) design options
  - Discuss CPU and Cache performance

  - Describe hardware multithreading techniques
- Memory Hierarchy
  - Identify the memory technologies found in computer processors, and computing systems
  - Describe the various ways of organizing memory and the impacts on cost-performance tradeoffs, speed, capacity, latency, and volatility (also include long term storage with tape drives, hard drives, and SSDs with performance enhancements like RAID.)
- Memory Model
  - Explains how high-level languages structure memory into stack, static, and dynamic regions and explain how each is used to include Mapping logical addresses to physical memory chips
- Digital logic, digital systems, and digital design
  - Design a simple combinational circuit using logic gates
  - Apply Boolean functions, algebraic theorems, and Karnaugh Maps to simplify combinational circuits
  - Explain the role feedback plays in sequential circuits
  - Identify and analyze common combinational and sequential circuits used in computer systems.
  - Language hierarchy
  - Explain how programming language abstractions at multiple levels are translated to lower levels and are executed
- I/O
  - Explain how interrupts are used to implement I/O control and data transfers.
  - Identify various types of buses in a computer system and describe how devices compete for a bus and are granted access to the bus.

Topical Description

Module	Topic
0	Course Policies and Overview QA Introduction to computing systems
1	CPU and Instruction Set Architectures Memory, IO Binary Decisions
2	BaseX numbering systems Binary Math
3	Digital Logic, AND,OR,NOT,NAND, NOR
4	Truth Tables Circuit Design Overview
5	Boolean Algebra K-Maps Circuit Design
6	Hardware: IC, MC
7	Adders, subtractors, multipliers, dividers Hardware: flip-flops, multi-vibrators, shift registers
8	Assembly Language

Notes to Instructors

- none