Syllabus

CSC 249 Computer Architecture And Organization

General Information

Date

July 31st, 2018

Author

April Devaux

Department

Computing Sciences

Course Prefix

CSC

Course Number

249

Course Title

Computer Architecture And Organization

Course Information

Credit Hours

4

Lecture Contact Hours

4

Lab Contact Hours

0

Other Contact Hours

0

Catalog Description

This course is designed for Computer Science majors. Topics include: classical von Neumann machine, major functional units, primary memory, representation of numerical(integer and floating point) and nonnumerical data, CPU architecture, instruction encoding, fetch-decode-execute cycle, instructional formats, addressing modes, symbolic assembler, assembly language programming, handling of subprogram calls at assembly level, mapping between high level language patterns and assembly/machine language, interrupts and I/O operations, virtual memory management, and date access from magnetic disk.

Key Assessment

This course does not contain a Key Assessment for any programs

Prerequisites

CSC 190

Co-requisites

None

Grading Scheme

Letter

First Year Experience/Capstone Designation

This course DOES NOT satisfy the outcomes applicable for status as a FYE or Capstone.

SUNY General Education

This course is designated as satisfying a requirement in the following SUNY Gen Ed category

None

FLCC Values

Institutional Learning Outcomes Addressed by the Course

Vitality
Inquiry
Perseverance

Course Learning Outcomes

Course Learning Outcomes

1. Recall the internal organization of computers, CPU, memory unit and Input/Outputs and the relations between its main components.
2. Analyze cost performance and design trade-offs in designing and constructing a computer processor including memory.
3. Perform elementary quantitative performance evaluation of computer systems.
4. Solve problems by assembly language programming.

Outline of Topics Covered

STRUCTURED COMPUTER ORGANIZATION

I. Languages, Levels, and Virtual Machines, Contemporary Multilevel Machines and Evolution of Multilevel Machines

- A. Classical Von Neumann machine
  - 1. MILESTONES IN COMPUTER ARCHITECTURE
- B. PROCESSORS
- C. CPU Organization
- D. Instruction Execution
- E. RISC versus CISC
- F. Design Principles for Modern Computers
- G. Instruction-Level Parallelism
- H. Processor-Level Parallelism
- I. PRIMARY MEMORY
- J. SECONDARY MEMORY
- K. INPUT/OUTPUT

II. GATES AND BOOLEAN ALGEBRA

- A. Gates
- B. Boolean Algebra
- C. BASIC DIGITAL LOGIC CIRCUITS
- D. MEMORY
- E. CPU CHIPS AND BUSES
- F. EXAMPLE CPU CHIPS
- G. EXAMPLE BUSES
- H. INTERFACING

III. AN EXAMPLE MICROARCHITECTURE

- A. The Data Path
- B. Microinstructions
- C. Microinstruction Control: The Mic-
- D. AN EXAMPLE ISA: IJVM
- E. AN EXAMPLE IMPLEMENTATION
- F. DESIGN OF THE MICROARCHITECTURE LEVEL

- F. THE ASSEMBLY PROCESS
    - 1. Fetch-Decode-Execute cycle
    - 2. LINKING AND LOADING
- G. A Small Assembly Language Program
- H. THE PROCESSOR
- I. MEMORY AND ADDRESSING
- J. THE INSTRUCTION SET
- K. THE ASSEMBLER

VII. THE TRACER

- A. WRITING PROGRAMS IN ASSEMBLY
    - 1. Subprogram Calls

VIII. ON-CHIP PARALELLISM

- A. Instruction-Level Parallelism
- B. On-Chip Multithreading
- C. Single-Chip Multiprocessors
- D. COPROCESSORS
- E. SHARED-MEMORYMULTIPROCESSORS
- F. MESSAGE-PASSING MULTICOMPUTERS
- G. GRID COMPUTING