

Prepared by the Department of Mathematics

Date of Departmental Approval: March 20, 2017

Date approved by Curriculum and Programs: April 12, 2017

Effective: Fall 2017

1. **Course Number:** CSC240
Course Title: Introduction to Computer Systems
2. **Description:** Students investigate how computer systems execute programs, store information, and communicate in order to become more effective programmers in terms of performance, portability, and robustness. Topics covered include: machine-level code and its generation by optimizing compilers, performance evaluation and optimization, computer arithmetic, memory organization and management, networking technology and protocols, and supporting concurrent computation. 4 class hours.
3. **Student Learning Outcomes (instructional objectives, intellectual skills):** Upon successful completion of this course, students are able to do the following:
 - Apply a deep understanding of systems-level issues to compiler, network, operating system, and computer architecture concepts.
 - Develop the intellectual tools needed to debug programs; in particular, by analyzing what happens with the hardware and software components of the system when a program runs.
 - Discuss the interactions between the hardware, operating system, compiler, and network.
 - Develop and evaluate low-level programs (C and assembly language) to investigate system issues.
 - Create high-performance programs that use computer hardware effectively.
 - Analyze software systems written by others.
 - Compare and contrast C as a procedural language and Java as an object-oriented language.
 - Demonstrate thread concepts, often by programming, including thread scheduling, the difference between processes and threads, concurrency, and synchronization.
 - Demonstrate data representation and memory management and explain the effects on program performance.
 - Solve problems involving computer arithmetic.
 - Explain caching, latency hiding, and scalable concurrency.
 - Analyze program performance and apply basic strategies for performance improvement.
 - Combine assembly code and high-level language code in one program.
4. **Credits:** 4 credits
5. **Satisfies General Education Requirement:** No
6. **Prerequisite:** CSC130 (Computer Programming II: JAVA)
7. **Semester(s) Offered:** Varies
8. **Suggested General Guidelines for Evaluation:** Comprehensive final examination, hour tests, programs, quiz papers, and homework papers
9. **General Topical Outline (Optional):** Please see the attached course outline.

CSC240 Introduction to Computer Systems

- I. Representing and Storing Information
 - A. Information Storage: Bits and Bytes
 - B. Integer Representation and Arithmetic
 - C. Floating Point
- II. Basics of C Programming Language
- III. C Structures and Pointers
- IV. C Dynamic Data Structures
 - A. C Libraries, Macros, and Header Files
 - B. Dynamic Libraries
- V. Loading and Linking a C Program
- VI. Machine Organization
 - A. Hardware Organization of System
 - B. Running a C Program on System
- VII. Machine-Level Representation of Programs
 - A. Historical Perspective
 - B. Program Encodings
 - C. Data Formats
- VIII. Processor Architecture
 - A. Programmer-Visible State
 - B. Assembly Language Instructions
 - C. Instruction Encoding
- IX. Loaders and Linkers
- X. Exceptional Control Flow
- XI. The Memory Hierarchy
 - A. Abstract Classes
 - B. Interfaces
 - C. Packages
- XII. Virtual Memory
- XIII. Network Programming
- XIV. Concurrent Programming