

CSC-Computer Science

CSC101 - Personal Productivity Software

This course provides a structured laboratory experience designed to develop and enhance a student's proficiency in using selected Windows microcomputer application software packages.

CSC120 - Problem Solving and Programming Constructs

This course will provide the student with a basic literacy of computers; present problem-solving heuristics and structured programming techniques; present language independent data types, operations, programming constructs and statements; introduce arrays and linked lists; and implement fundamental programs using an appropriate programming language.

CSC124 - Computer Programming I

This course builds on CSC 120. It gives the student a thorough understanding of the presently adopted language so that the student will develop the ability to program in the language. Emphasis is placed on efficient software development using structured programming techniques. Students are required to write, test and run programs.

CSC201 - Internet Concepts

This primarily hands-on course will review computer system concepts, will introduce the student to all facets of the Internet, and will develop a student's proficiency in Web page design and publishing.

CSC216 - Logic and Switching Theory of the Computer

This course provides the student with an in-depth study of the basis of digital computers. Number systems, arithmetic operations, codes, Boolean algebra, Boolean minimization techniques, state transition tables and state

Course Descriptions

transition graphs are discussed. Extensive emphasis is placed on the analysis and synthesis of synchronous and asynchronous combinational networks which form digital computers.

CSC265 - Object-Oriented Programming

Object Oriented Programming- This course teaches object-oriented programming. Object-oriented programming offers a natural method for designing software systems that build on the concepts of data abstraction, information hiding and modularity. Students will design and implement solutions to problems using an object-oriented programming language.

CSC302 - Visual Programming

This course teaches Windows applications programming using the object-oriented event-driven programming paradigm, with the programming language VisualBasic.NET. It is designed as a beginning OOED programming course, but assumes students know Windows object vocabulary, have basic Windows file management skills, and are familiar with the generic procedural programming language constructs of decision structures and looping.

CSC304 - COBOL

This course introduces students to the essential elements of the COBOL language using well-structured programming techniques. Students will write and execute report programs, control break programs, data validation programs, programs that implement tables and sequential update programs. Good analysis, design and structure will be emphasized.

CSC306 - Fortran

The FORTRAN language will be studied. Most of the major programming constructs of FORTRAN will be covered, including assignment statements, loops, decisions, subprograms, arrays, character manipulation and file processing. Comparisons with other languages will be made, and documentation of programs will be emphasized.

CSC308 - Python

This course enables the student to acquire a thorough understanding of the Python language and its application in solving real world problems. Emphasis is placed on efficient software development using structured programming techniques, Object Oriented Programming, GUI interfaces, as well as a variety of Python modules and packages. Students are required to design, write, test, and run programs using an appropriate version of Python.

CSC322 - Data Base Application Development

Building upon the conceptual understanding of a modern DBMS (Database Management System) and database and table design concepts gained in CIS 321 – Database Management Systems and Design, this course provides students with the practice of applying database technology via the Oracle DBMS to the solution of business and other information-related problems. Experience is provided with database design and implementation based on a thorough analysis of requirements and information modeling. The use of Structured Query Language (SQL) for interaction with a working DBMS for data creation, manipulation and extraction is stressed as well as optimization techniques, such as view creation and indexing. PL/SQL and database triggers are introduced.

CSC323 - Assembly Language Programming

In this course students will study assembly language. In doing so, students will develop some concepts related to the architecture and operations of the computer. Programs will be written and implemented using the instructions in this assembly language. Constructs such as selection, looping and subprograms will be implemented.

CSC328 - Data Structures

The design, use and programming of data structures, such as stacks, queues, linked lists and binary trees, will be discussed. Sorting and searching methods are also discussed in this course. The analysis of algorithms will be considered as well as the applications of the various data structures.

CSC352 - Global, Economic and Social Ethical Issues in Computing

This course covers issues related to various global, economic and social frameworks and moves to topics specifically related to computers. Emphasis is placed on the study of ethical situations that arise as a consequence of the development and deployment of computers and related technologies, and also from parties with malicious intents toward prevalent technologies. Topics can include areas such as: security, economics of information systems, computer crime and hacking, computer software ownership, privacy, risks of computing, professional liability, internet freedom in computing and international laws and governance. The course is to be delivered in a writing intensive format, with treatise and arguments communicated effectively to a wide variety of audiences.

CSC360 - Analysis of Algorithms

This course covers algorithm analysis theory and techniques. Students learn properties of both efficient and inefficient algorithms. The importance of analyzing algorithms before implementing them will be emphasized. This course will teach the skills necessary to determine the best algorithm for a given problem. We will investigate greedy, graph theoretic, divide and conquer, and distributed algorithms. We will cover both polynomial time algorithms and NP-completeness.

CSC378 - Computer Architecture

This course provides the student with an in-depth study of the organization of the central processing unit, arithmetic logic unit, control unit, instruction formats, and addressing schemes of digital computers. Extensive emphasis is placed on the translation of assembly language instructions into their micro-sequence operations within the control unit and the interconnection and control of registers, arithmetic logic units, memory units, and busses which form the central processing unit and the digital computer.

CSC400 - Operating Systems

This course involves an introductory study of the main elements of an operating system – memory management, process management, device management, and file management. An operating system defines an abstraction of hardware behavior with which programmers can control the hardware. It also manages resource sharing among the computer's users. This course investigates these concepts as well as issues that influence the design of contemporary operating systems, including management of processes, memory, devices, and files. Additional special topics may include scripting, security, fault tolerance, and real-time systems.

CSC419 - Internship

This course is designed for the computer science major who is seeking work experience in the computer science area. This intern experience will enable the student to apply her/his knowledge of computers in the real work-place. The internship will provide the student with the valuable computer experience that should enhance the student's job opportunities upon graduation.

CSC420 - Artificial Intelligence

This course offers a selective survey of key concepts and applications of artificial intelligence and an introduction to a language commonly used for building AI systems.

CSC424 - Numerical Analysis

Numerical Analysis-In this course, various mathematical algorithms and applications relating to the numerical computation are investigated. Topics include: roundoff errors and computer arithmetic; numerical instability; error analysis and estimation; approximation; Gaussian elimination and pivoting strategies for linear systems; numerical integration and numerical solution of differential equations; curve fitting, polynomial approximation; and regression.

CSC455 - Structures of Programming Languages

Students will study the four categories of programming languages: imperative, object-oriented, functional and logic. An in-depth discussion of the imperative languages will be followed by discussions of the other three paradigms. Students will be required to investigate at least one language.

CSC460 - Language Translation

This course studies the design and construction of compilers. Lexical analysis, syntactic analysis and code generation are investigated in detail. Language design, interpreters, semantic analysis, intermediate code generation and code optimization are also considered.

CSC475 - Theory of Languages

This course is an introduction to abstract machine theory, combinatorial systems, computable functions, and formal linguistics. Topics include finite-state machines, regular sets, Turing machines, Chomsky hierarchy grammars and languages. Emphasis is on surveying basic topics and developing an intuitive understanding in the theory of languages.

CSC490 - Senior Project I: Software Engineering

This course introduces students to software engineering. They will study its history, terminology, requirements, specifications and design. The students will write requirements, specifications and design documents and one or more papers on software engineering topics.

CSC492 - Senior Project II

This course is a continuation of the Senior Project I: Software Engineering course and the capstone course of the program. The project proposal developed and designed in the first Senior Project class will be implemented in

Course Descriptions

this course. The student will produce a project users' manual and will demonstrate proficiency in the academic program through the development of the project.