

Computer Science

CSCI 5. Computer and Applications (3)

An introduction to the computer: tools, applications, and graphics. Overview of the components of computer systems; discussion on software systems, electronic mail, influence of computers on society and the future of computing; extensive hands-on experience with application tools and programming. PC (Windows) environment. *CR/NC* grading only. (2 lecture, 2 lab hours)

CSCI 7. Computer Literacy (3)

Overview of the history of computing, a presentation of the components of computer hardware and software systems, a study of applications, programming, electronic mail, societal impact, and the future of computing. Macintosh environment. (2 lecture, 2 lab hours)

CSCI 15. C and C++ Programming (2)

Prerequisite: programming experience in a major high-level language, e.g., BASIC, COBOL, FORTRAN, Pascal. An introduction to the C and C++ programming languages. Types, operators, expressions, flow of control, functions, pointers, and arrays. Standard libraries and programming tools. Emphasis on programming projects.

CSCI 30. Introduction to the Internet (3)

Topics include e-mail, Web browsers, searching, evaluation of Web resources, HTML, Web-page design, encryption, and basic network communication. Special emphasis is placed on the underlying technologies. (2 lecture [1 traditional, 1 online], 2 lab hours)

CSCI 40. Introduction to Programming and Problem Solving (4)

Prerequisites: Students must take the ELM exam; students who do not pass the exam must record a grade of C or better in a college-taught intermediate algebra course; trigonometry. Introduction to problem solving, algorithm development, procedural and data abstraction; program design, coding, debugging, testing, and documentation; a high-level programming language. (3 lecture, 2 lab hours)

CSCI 41. Introduction to Data Structures (4)

Prerequisite: CSCI 40. Programming methodology, program correctness. Review of data types. Data structures: linear and nonlinear structures, files. Implementation of data structures. Recursion. Searching and sorting. (3 lecture, 2 lab hours)

CSCI 60. Foundations of Computer Science (4)

Prerequisites: CSCI 40 (may be taken concurrently.) Abstraction, iteration, induction, recursion, complexity of programs, data models, and logic. (3 lecture, 2 lab hours)

CSCI 101. Computational Foundations for Bioinformatics (3)

Prerequisite: CSCI 1, BIOL 102. Computational approaches to problems in molecular biology; Algorithms, heuristics, strings, and graphs. Sequence comparison and multiple alignment. Selected topics such as scripting, visual programming, laboratory workflow, databases, and queries. (2 lecture, 2 lab hours)

CSCI 105T. Workshop on Computer Languages (1-3; max total 6)

Prerequisite: CSCI 40 or permission of instructor. Workshops in the use of various high-level programming languages or other selected languages in areas of database, statistical computation, or operating systems.

CSCI 112. Introduction to Computer Systems (4)

Prerequisite: CSCI 41, 60. Computer arithmetic. Von Neumann architecture. Instruction sets, data types, formats, addressing. Register and ALU organization. Memory hierarchy. I/O. Bus organization. Study of one or more assembly languages. Basics of implementation of higher-level languages. (3 lecture, 2 lab hours)

CSCI 113. Introduction to Computer Organization (4)

Prerequisite: CSCI 41. Fundamental issues of computer design at register-transfer level. Logical design of basic combinational and sequential modules. Organization and design of major functional blocks: ALU, CPU, memory, cache, input/output, hard-wired and microprogrammed control. Simulation of computer organization. Introduction to high-performance superscalar computer organization. (3 lecture, 2 lab hours)

CSCI 115. Algorithms and Data Structures (4)

Prerequisites: CSCI 41, 60; MATH 75. Review of basic data structures. Graph, search paths, and spanning trees. Algorithm design and analysis of sorting, merging, and searching. Memory management, hashing, dynamic storage allocation. Integration of data structures into system design. (3 lecture, 2 lab hours)

CSCI 117. Structures of Programming Languages (4)

Prerequisites: CSCI 41, 60. General concepts and paradigms of programming languages; scope and binding rules, applications and implementations of language concepts. Languages selected from: ADA, ICON, Miranda, ML, MODULA 2, OCCAM 2, PROLOG, LISP, Scheme, Smalltalk. (3 lecture, 2 lab hours)

CSCI 119. Introduction to Finite Automata (4)

Prerequisites: CSCI 41, 60. Strings, languages, and fundamental proof techniques. Regular expression, regular grammar, regular languages, finite automata, their interrelationship, and their properties. Introduction to context-free languages. (3 lecture, 2 lab hours)

CSCI 124. Introduction to File Processing (3)

Prerequisite: CSCI 115. Definition of file components, access methods, and file operations. Algorithms for efficient implementation of data structures; characteristics of bulk storage media for mainframe and microcomputers. Introduction to database management systems.

CSCI 126. Database Systems (3)

Prerequisite: CSCI 124. Database concepts; hierarchical and relational network models; object-oriented data models. Data normalization, data description languages, data manipulation languages, and query design.

CSCI 130. Web Programming (3)

Prerequisite: CSCI 115. Programming for the World Wide Web. Web servers and clients, Internet and Web protocols, and mark-up languages. Client-side scripting, including both gateway and filter-based approaches. (2 lecture, 2 lab hours)

CSCI 134. Compiler Design (3)

Prerequisites: CSCI 112, 115, 119. Syntax and semantics of programming languages. Lexical analysis, parsing techniques, parser generator, SLR and LALR parsing. Introduction to symbol table organization and semantic routines. Compiler generators.

CSCI 144. Introduction to Operating Systems (4)

Prerequisites: CSCI 41 and CSCI 112 or ECE 118. Operating system history and services. File systems. Memory management. Process management — concurrent processes, communication, semaphores, monitors, deadlocks. Resource management