Computer Science - Bachelor's Degree for Professionals

Bachelor of Science in Computer Science

48 credits minimum; 36 upper division credits. 2.0 minimum GPA.

CSC 210	Introduction to Computer Programming	4
CSC 220	Advanced Computer Programming	4
CSC 335	Software Engineering	4
CSC 340	Operating Systems	4
CSC 350	Data Communication & Networks	4
CSC 499	Capstone	4
Additional Upper Division Computer Science Credits		20
MATH 241	Discrete Mathematics	4
Total Hours		48

Minor in Computer Science

20 Computer Science credits minimum; 12 credits upper division.

Courses

Lower Division

CSC 102. Introduction to Computers. (4).
A first course intended for novice computer users that introduces microcomputers, word processing, spreadsheets, selected computer applications software and Internet utilities. In addition, the student learns the proper use of various computer peripherals including diskette drives, mice, keyboards, scanner and advanced digital devices.

CSC 110. Concepts of Programming. (4).
Introduction of logic concepts in programming. Breadth approach to essential elements of computer programming. Text based operating systems such as DOS will be discussed. Topics covered are problem solving concepts, computer systems, disk operating systems, computer programming languages, programming fundamentals, testing and debugging, conditions and branching, loops, flowcharts, compound statements, non-compound statements, top-down program design.

CSC 205. Programming for Scientists. (4).
This course introduces the principles of computer programming, problem-solving methods, and algorithm development from a scientific perspective. The programming languages covered are C (a compiled language popular among engineers and mathematicians), and Perl (a scripting language popular among bioengineers) both in wide use in scientific fields. Also covered are introductory software engineering techniques and tools necessary to convert a functional specification to a properly functioning program. Examples and assignments will be drawn from the natural sciences. (Cross listed with SCI 205).

CSC 210. Introduction to Computer Programming. (4).
First-semester computer programming course. This course introduces the principles of computer science, problem-solving methods and algorithm development using a high-level language. This is a programming class primarily for computer science, computer information systems, mathematics, and science majors. The ability to use a computer is essential. Prerequisites: CSC 110 or permission of instructor, MATH 110 or equivalent.

CSC 220. Advanced Computer Programming. (4).
A second-semester computer programming course. This course takes a state-of-the-art approach to software design/development with object-oriented techniques. Topics include algorithm analysis, string processing, internal search/sort methods, complex data structures, design strategies, and code reusability. Prerequisite: CSC 210.

CSC 282C. ST: Select Topic (core). (1-4).
Select Topic approved for core.

Upper Division

CSC 300. Visual Programming. (4).
Advanced programming course which focuses on the design of visual user-interface in the Windows environment. Topics include basic forms, simple structures, variables, control mechanism, types and expressions, complex data structure, looping, functions, procedures, selections, multiple forms, files and arrays. Prerequisite: CSC 210.

CSC 310. Algorithms. (4).
Continues the study of the design and analysis of algorithms, particularly those handling complex data structures and non-numeric processes. Includes an introduction to algorithm design techniques, algorithm verification and the impact of parallel computation on algorithms, operating systems and architectures. A brief introduction is given to artificial intelligence focusing on data representation and heuristic search methods. Prerequisites: CSC 210, MATH 241.

CSC 315. Object-Oriented Design and Analysis. (4).
Discusses the features and advantages of an object-oriented approach to problem solving. Topics include abstraction, inheritance, polymorphism, object-oriented design, analysis, implementation and testing. Prerequisites: CSC 210.

CSC 321. Computer Organization and Architecture. (4).
Principles of computer organization and architecture are introduced from a layered point of view, beginning at data representation and progressing through the machine language execution cycle. Representative software-hardware tradeoffs in the implementation of various computer system components will be presented. The design and interface to a variety of peripheral devices will also be discussed. The emphasis will be on the hardware aspects of a computer system. Prerequisites: CSC 210, MATH 241.

CSC 322. Introduction to Robotics. (4).
An introductory study of the field of robotics-devices designed and programmed to perform various tasks. Topics include; hardware design (mechanical and electronic); software design; power subsystems; sensors; actuators; effectors; applications; comparison to biological systems; safety; societal impact and ethics. Student will study theory (lecture component) and build/program a robot (laboratory component).

CSC 325. Organization of Programming Languages. (4).
Covers introduction of major language histories, common components, built-in structures, compositions of basic structures, language specification, analysis techniques, runtime behavior, de-facto standards, and future developments. Prerequisites: CSC 210, MATH 241.

CSC 331. Systems Analysis. (4).
This is the first course in system engineering that stresses the system development life cycle. Students learn ways of organizing the structure and process of building very large-scale systems that may or may not involve computers. Includes information gathering, design tradeoffs, implementation strategies, product liability, acceptable risk analysis and project follow-up. Prerequisites: CSC 210, MATH 241.

CSC 332. Introduction to E-Commerce. (4).
Overview of eCommerce from business aspects to required eCommerce technical skills. A lecture based course with extensive online research for eCommerce information, useful sites, case studies and Web tools. A basic e-Commerce architecture of three tiers such as the front-end tier, the Web server tier and the back-end system tier in Windows NT and Unix. Connectivity to the back-end database system and legacy systems. Security, protection, electronic payment, firewall and proxy. Several Web designing tools and programming skills. The course builds a foundation for students to pursue higher level e-Commerce courses. Prerequisites: CSC 110 or permission of instructor.

CSC 335. Software Engineering. (4).
Presents a formal approach to state-of-the-art techniques for software design and development, involving students in a team approach to organizing, managing and developing software. Prerequisites: CSC 210, MATH 241.

CSC 340. Operating Systems. (4).
Discusses the major functionality and principles behind all major operating systems tasks, including user interface, hardware sharing among users, data sharing among processes, user protections, resources scheduling among users, multi-user environment, multi-processing and real-time systems. Prerequisites: CSC 210, MATH 241.

CSC 344. Web Design. (4).
Studies the backbone of dynamic Web documents. Subjects include Web design standards, and Web-based application programming to make layout, tables, style sheets, templates, libraries, frames and rollovers. HTML and script languages such as Java Scripts, GUI design paint tools and plug-ins are studied in depth. Prerequisites: CSC 210, MATH 241.

CSC 350. Data Communication & Networks. (4).
Includes discussion of distributed data processing, communication techniques, wide-area and local-area networks, integrated services digital network, open-systems interconnection, security and network management. Prerequisites: CSC 210, MATH 241.

CSC 355. Client/Server Fundamentals. (4).
Discusses modern technology in network communication and cooperative computation. Topics include discussion of client/server design concept, software expectation, hardware requirement, service, support and training issues. Prerequisites: CSC 210.

CSC 360. Computer System Security. (4).
An introduction of security issues in computer system and data communications, including Data Encryption Standard, public-key systems, digital signatures, ciphers, data compression, data manipulation and supporting techniques. Prerequisites: CSC 210, MATH 241.

CSC 370. Multimedia Technology. (4).
Introduces modern multimedia technologies. Topics include basic concepts, principles, sound, image, animation, standards, hardware and software requirements, new technologies, current research and practice, and future directions. Prerequisites: CSC 210, MATH 151.

CSC 400. Graphical User Interface. (4).
An introductory course to user interface design fundamentals. Topics include development methodologies, evaluation techniques, user-interface building tools, considerations in the design phase, identification of applicable design rules, and successful delivery of the design. Prerequisite: CSC 210.

CSC 405. Graphics. (4).
Review of graphic display architecture and graphic input devices. Coverage includes two- and three-dimensional drawing, viewing, clipping, transformations, shading and data structures for graphics systems. Prerequisites: CSC 210, MATH 241.

CSC 410. Database Management Systems. (4).
Studies the concepts and structures necessary to design and operate a database management system. Topics include data modeling, relational database design, and database querying. Prerequisites: CSC 210, MATH 241.

CSC 412. Bioinformatics-Computational. (4).
The course introduces commonly used methods for analyzing biological data such as DNA and protein sequences and covers phylogenetic tree construction and 3D folding of biomolecules. It examines bioinformatics algorithms such as sequence search and alignment and its underlying principles and implements simple algorithms using Perl programming language. Prerequisites CSC 110 / CSC 210, & MATH 352 for CS majors. CSC 110 / CSC 210, MATH 352, & Biol 422 for Biology majors.

CSC 435. Competition Problem Solving. (4).
This course challenges students in real-world problem solving and prepares for the prestigious Association of Computing Machinery International Collegiate Programming Contest (ACM-ICPC). What is a good problem solving? It can be one that delivers the correct result. What is then a good competition problem solving? It is one that delivers the correct result in the expected amount of computation time. While there are a few classic problem-solving approaches, students in this course will learn to devise techniques in a deeper manner because most real-world problems command new approaches instead of the mere application of classic ones. Such skill will be of great importance for future advancement in both the industry as well as the academics.

CSC 482. Selected Topics. (1-4).

CSC 482C. ST: Select Topic (core). (1-4).
Select Topic approved for core.

CSC 490. Independent Study. (1-4).

CSC 492. Internship. (1-4).
(graded P/NC only).

CSC 493. Field Study. (1-2).

CSC 496. Directed Research. (1-3).

CSC 499. Capstone. (4).
Undergraduate research or development project. The exact nature of the project is negotiated with the sponsoring professor.