

COMPUTER SCIENCE (CSCI)

CSCI 212. Computer Programming Principles. (2 Credits)

A gentle introduction to computer programming using the principles of algorithmic thinking. Intended for students who intend to take CSCI 235 in a future semester but desire a slower-paced first experience with programming. Course is offered occasionally.

CSCI 232. Programming for the Mathematical Sciences. (4 Credits)

An introduction to computer programming for students who want to apply programming to mathematics or the natural sciences. Topics include problem-solving, types, control structures, and functions. Programming is taught using Python or a similar language and libraries for statistics and numerical processing. Not open to students who have passed CSCI 235 or the equivalent.

CSCI 234. Programming for Business Analytics. (4 Credits)

This course will provide an introduction to programming using Python or a similar language for students who have little or no prior experience in programming. Topics include basic problem-solving techniques, algorithm development, decision and repetition structures, input and output functions, file handling, basic data structures, and introduction to object-oriented programming. Not open to students who have passed CSCI 235 or the equivalent.

Tags: AAQR

CSCI 235. Programming I: Problem Solving. (4 Credits)

A first course in computer programming for beginners. Structured and object-oriented programming in Java or a similar programming language. Types, control structures, methods, and recursion; objects, classes, interfaces, encapsulation and polymorphism; exceptions, library classes, file I/O, linked lists, and graphical user interfaces.

Tags: AAQR

CSCI 236. Accelerated Introduction to Programming. (2 Credits)

A condensed, accelerated version of CSCI 235 for students with prior programming experience. This course focuses on preparing students to take CSCI 245. Offered occasionally.

CSCI 243. Discrete Mathematics and Functional Programming. (4 Credits)

Sets, logic, the nature of proof, induction, algorithms, algorithm correctness, relations, lattices, functions, and graphs. Functional programming and recursion using SML or a similar programming language. May not be taken after MATH 341 or MATH 351.

Tags: AAQR

CSCI 245. Programming II: Object-Oriented Design. (4 Credits)

A gateway to the computer science major, introducing a range of themes in the field of computer science. Object-oriented programming in Java or a similar programming language: code reuse with composition and inheritance; generic types; design patterns. Software development: development tools, attributes of good design. Algorithmic analysis; searching and sorting algorithms. Abstract data types: stacks, queues, trees, hashing; linked vs array-based implementation. Systems programming in C; pointers and dynamic allocation; model of machine memory, organization, and execution. Prerequisite: CSCI 235 or departmental approval.

CSCI 335. Software Development. (4 Credits)

Principles and practices of software development including design patterns, validation and testing, coordination of team projects. Introduction to databases and user interface design. Professional issues in computing. Prerequisite: CSCI 245; Pre or Corequisite: CSCI 243.

CSCI 345. Data Structures & Algorithms. (4 Credits)

Formal and experimental approaches to verifying algorithms' correctness and analyzing their efficiency. Abstract data types and their implementations. Efficient implementations of maps using balanced binary search trees and hash tables. Graph algorithms. Dynamic programming. Prerequisites: CSCI 243 and CSCI 245.

CSCI 351. Introduction to Computer Systems. (4 Credits)

An introduction to low-level systems issues from the perspective of the programmer. Representation of both data and program as produced by a compiler; hardware support for memory, input/output, and parallelism; fundamental ideas in operating systems and networking. Prerequisite: CSCI 245.

CSCI 357. Networking. (4 Credits)

Examination of the fundamental problems in computer internetworking, from the link to application levels, with particular attention to the Internet protocols. Issues include naming/addressing, error-handling, routing, and decentralized control. Prerequisite: CSCI 351. Course offered alternate years.

CSCI 359. Information Security. (4 Credits)

An introductory course in the technical aspects of information security, exploring both theory and practice. Basic concepts and policies: confidentiality, integrity, availability, access control. Cryptography and its applications: ciphers, hashes, key exchange, authentication, blockchain. Topics in systems and network security: malware, botnets, covert channels, denial of service, firewalls, Web security, anonymous communication. Prerequisite: CSCI 351.

CSCI 361. Computer Graphics. (4 Credits)

Introduction to graphical programming environments, OpenGL libraries. Rendering three-dimensional images, transformations, windowing, clipping, shading, and image enhancements. Prerequisite: CSCI 345. Course is offered occasionally.

CSCI 365. Programming Language Concepts. (4 Credits)

A survey of the design and implementation of programming languages: grammars, parsing, and abstract syntax; compilers, interpreters, and other language systems; type-checking and other static analyses; formal semantic specifications; implementation of imperative, functional, and object-oriented language features; type-soundness proofs; tail form and continuation passing style. Course is offered occasionally. Prerequisite: CSCI 335 and CSCI 351.

CSCI 371. Database Management Systems. (4 Credits)

An introduction to database management systems. Topics include history and motivation for database systems, relational model and algebra, database design, SQL overview, schema refinement and normal forms, data warehousing concepts and modeling, basics of database administration, storage of data on disk systems, file organization, hash and tree indexing, and web-based access of database systems. Prerequisite: CSCI 335 and CSCI 345.

CSCI 373. Platform-Specific Development. (4 Credits)

Introduction to developing software on a specific platform, such as iOS programming using Swift for applications on iPhones and iPads. Topics include developer tools, programming in a platform-specific language, MVC design pattern, user interfaces, persistent storage solutions, and frameworks for additional app functionality. Prerequisite: CSCI 335. Course offered alternate years.

CSCI 381. Machine Learning. (4 Credits)

Theory, algorithms, and applications of machine learning. Machine learning techniques including k-nearest neighbors, expectation-maximization, neural nets, support vector machines, and principal component analysis. Ethical considerations for how machine learning applications are used and how they affect society. Prerequisite: CSCI 345 and MATH 245. Course offered alternate years.

CSCI 384. Computational Linguistics. (4 Credits)

An exploration of the theory, algorithms, and applications of natural language processing. Traditional approaches such as n-gram language models, information theory, and hidden Markov models, as well as modern approaches such as recurrent neural nets and large language models. Applications of computational linguistics such as part-of-speech tagging, authorship attribution, automatic translation, sentiment analysis, and text generation. Prerequisite: CSCI 345.

CSCI 394. Seminar. (2 or 4 Credits)

Selected topics in Computer Science at each offering, including such subjects as object-oriented design, e-commerce, human computer interface, networking services. May be taken again when a different topic is offered. Prerequisite: Departmental approval.

CSCI 445. Analysis of Algorithms. (4 Credits)

An introduction to algorithmic efficiency and to techniques for the design and analysis of efficient algorithms. General topics include review of asymptotics, algorithm design techniques (such as divide-and-conquer, dynamic programming, and greedy algorithms), graph algorithms, languages and automata, and NP-completeness. Prerequisite: CSCI 345. Course offered alternate years.

CSCI 455. Operating Systems. (4 Credits)

Design and implementation of operating systems. Processes, threads, and concurrency; physical and virtual memory management; disk, files, and filesystem implementation; the boot process; input and output systems; support for virtualization; protection and security. Course offered alternate years. Prerequisite: CSCI 335 and CSCI 351.

CSCI 493. Mentored Research Seminar. (2 or 4 Credits)

Faculty and student collaboration on a project of mutual interest. Limited enrollment - faculty approval required.

CSCI 494. Social and Ethical Issues in Computing. (2 Credits)

A study of the ways in which the computer and communications revolution is changing society to develop an awareness of and sensitivity to the ethical issues that arise in computer science and related professions. Prerequisite: Senior standing in the major.

General Education: SHAR

CSCI 495. Independent Study. (1 to 4 Credits)

An individually adapted study of any aspect of computing science or its relationship to other fields of study.

CSCI 496. Internship. (2 or 4 Credits)

Graded pass/fail. Prerequisite: junior or senior standing with Computer Science major. May repeat once for a total of 4 hours.