Undergraduate students may choose from one of two degree options: the B.S. in computer science, which is intended for students interested in the design and implementation of software systems, or the B.S. in information technology, which focuses on the application of existing computing technologies to meet the information needs of organizations and individual computer users. The B.S. in computer science also offers a bioinformatics option, which is designed for students who wish to apply computer science expertise in the life sciences.

https://ceps.unh.edu/computer-science/

### Programs

- Analytics and Data Science Major: Analytics Option (B.S.) ([http://catalog.unh.edu/undergraduate/engineering-physical-sciences/programs-study/computer-science/analytics-data-science-major-analytics-option-bs](http://catalog.unh.edu/undergraduate/engineering-physical-sciences/programs-study/computer-science/analytics-data-science-major-analytics-option-bs))
- Computer Science Major (B.S.) ([http://catalog.unh.edu/undergraduate/engineering-physical-sciences/programs-study/computer-science/computer-science-major-bs](http://catalog.unh.edu/undergraduate/engineering-physical-sciences/programs-study/computer-science/computer-science-major-bs))
- Computer Science Minor ([http://catalog.unh.edu/undergraduate/engineering-physical-sciences/programs-study/computer-science/computer-science-minor](http://catalog.unh.edu/undergraduate/engineering-physical-sciences/programs-study/computer-science/computer-science-minor))
- Computer Programming Cognate ([http://catalog.unh.edu/undergraduate/engineering-physical-sciences/programs-study/computer-science/computer-programming-cognate](http://catalog.unh.edu/undergraduate/engineering-physical-sciences/programs-study/computer-science/computer-programming-cognate))
- Information Technology Major (B.S.) ([http://catalog.unh.edu/undergraduate/engineering-physical-sciences/programs-study/computer-science/information-technology-major-bs](http://catalog.unh.edu/undergraduate/engineering-physical-sciences/programs-study/computer-science/information-technology-major-bs))
- Information Technology Minor ([http://catalog.unh.edu/undergraduate/engineering-physical-sciences/programs-study/computer-science/information-technology-minor](http://catalog.unh.edu/undergraduate/engineering-physical-sciences/programs-study/computer-science/information-technology-minor))
- Information Technology Cognate ([http://catalog.unh.edu/undergraduate/engineering-physical-sciences/programs-study/computer-science/information-technology-cognate](http://catalog.unh.edu/undergraduate/engineering-physical-sciences/programs-study/computer-science/information-technology-cognate))

### Courses

#### Computer Science (CS)

**CS 401 - Computers and Their Applications**  
*Credits: 4*

Use of computers to manage and analyze information across a variety of settings and disciplines. Introduces major categories of computer software, including word processing, spreadsheets and database systems. Covers basic computer concepts and the computer’s role in today’s society. Significant hands-on work required outside of the class. Not open to CS majors. CEPS students should check with their major department for approval.  
*Attributes: Environment,TechSociety(Disc)*

**CS 401H - Honors/Computers & Their Appl**  
*Credits: 4*

Use of computers to manage and analyze information across a variety of settings and disciplines. Introduces major categories of computer software, including word processing, spreadsheets and database systems. Covers basic computer concepts and the computer’s role in today’s society. Significant hands-on work required outside of the class. Not open to CS majors. CEPS students should check with their major department for approval.  
*Attributes: Environment,TechSociety(Disc)*

**CS 404 - Do-It-Yourself Internet**  
*Credits: 4*

The objective of this course is to demystify the design process that leads to the evolution of the Internet. In doing so, we investigate the ways that technology changes to meet the needs of society, how society changes in response to these new technologies and how these societal changes create pressures that produce needs for new technologies.  
*Attributes: Environment,TechSociety(Disc)*

**CS 405 - Introduction to Applications Programming**  
*Credits: 4*

Introduces the concepts and techniques of computer programming. Particular emphasis on computer programming as a problem-solving technique for business applications. The basic software development process (modeling, algorithm design, programming, testing and debugging) is illustrated. CEPS students should check with their major department for approval. Not open to CS and IT majors.  

**CS 408 - Living in a Networked World: The Good, the Bad, and the Ugly**  
*Credits: 4*

The objective of this course is to explore the implications of living in a networked world. The course surveys the fundamental technologies and practices that make up the Internet and then ask the student to examine the ramifications of using the technologies. Users of the technologies should understand the technology in order to make educated decisions about how to use it safely and effectively. Students have the opportunity to self-publish by using various current technologies including blogs, discussion boards, email and creating web pages using xhtml.  
*Attributes: Environment,TechSociety(Disc)*

**CS 410 - Introduction to Scientific Programming**  
*Credits: 4*

Introduces the concepts and techniques of computer programming. Particular emphasis on computer programming as a problem-solving technique in science and engineering applications. Either the C language or Python is taught and used for assignments. Good programming style is stressed. Significant out-of-class programming required. Not open to students who have completed CS 407, 415, or the equivalent.
CS 410C - Introduction to Scientific Programming/C
Credits: 4
Introduces the concepts and techniques of computer programming. Particular emphasis on computer programming as a problem-solving technique in science and engineering applications. Good programming style is stressed. Significant out-of-class programming required. Not open to students who have completed CS 415 or equivalent.

CS 410P - Introduction to Scientific Programming/Python
Credits: 4
Introduces the concepts and techniques of computer programming. Particular emphasis on computer programming as a problem-solving technique in science and engineering applications. Good programming style is stressed. Significant out-of-class programming required. Not open to students who have completed CS 415 or the equivalent.

CS 414 - From Problems to Algorithms to Programs
Credits: 4
This course is an introduction to the design and implementation of computer programs. The basic software development process (modeling, algorithm design, programming, testing and debugging) is illustrated through problem examples. Programming techniques are introduced to allow students to implement and evaluate solutions as programs.
Attributes: Quantitative Reasoning(Disc)

CS 415 - Introduction to Computer Science I
Credits: 4
Theory and practice of computer science. Algorithm development and analysis; data abstraction techniques; elementary data structures; dynamic memory manipulation; debugging; and program design issues. Computer systems and applications. Intended for CS majors.

CS 416 - Introduction to Computer Science II
Credits: 4
Theory and practice of computer science. Algorithm development and analysis; data abstraction techniques; elementary data structures; dynamic memory manipulation; debugging; and program design issues. Computer systems and applications. Intended for CS majors. Prereq: CS 415.

CS 417 - From Programs to Computer Science
Credits: 4
Accelerated coverage of programming techniques for students with experience equivalent to CS410 or CS 414. Covers basic algorithm analysis. Topics include basics of classes, inheritance, and data abstraction; linear data structures (vectors, lists, stacks and queues); trees and simple graphs; hash tables; sorting and searching; recursion; and basic graph traversal algorithms. Numerous labs and programming assignments build skills in planning, problem solving, and debugging: this is a hands on course. Prereq: CS 410 or CS 414 or equivalent.

CS 501 - Professional Ethics and Communication in Technology-related Fields
Credits: 4
A mixed lecture/seminar course intended to improve both reasoning and ability to communicate effectively in front of an audience. Students learn basic forms of ethical argument, they read about ethical situations in which technology and technology professions play a key role, and they participate in student-led discussions about the reading. Students also make oral presentations about both ethical and technical topics, and evaluate each others' presentations in order to improve their sense for what makes a good presentation. Prereq: ENGL 401.
Attributes: Environment, TechSociety(Disc); Inquiry (Discovery)

CS 515 - Data Structures and Introduction to Algorithms
Credits: 4
Reviews basic data structures; advanced data structures such as graphs, B-trees, and AVL trees; abstract data structure design and programming techniques; use of data abstraction language. Introduction to algorithm analysis. Prereq: CS 416 or CS 417.

CS 520 - Assembly Language Programming and Machine Organization
Credits: 4
Assembly language programming and machine organization: program and data representation; registers, instructions, and addressing modes; assemblers and linkers. Impact of hardware on software and software on hardware. Historical perspectives. Prereq: CS 416 or CS 417.

CS 580 - Introduction to Topics in Computing
Credits: 1-2
Introductory material not normally covered in regular course offerings, but of value to students prior to internships or senior-level courses. May be repeated for credit.

CS #595 - Professional Ethics and Communication in Computer Science
Credits: 2
A seminar course intended to improve both reasoning and ability to communicate effectively in front of an audience. Students learn basic forms of ethical argument, they read about ethical situations in which technology and technology professions play a key role, and they participate in student-led discussions about the reading. Students also make oral presentations about both ethical and technical topics, and evaluate each others' presentations in order to improve their sense for what makes a good presentation.

CS 619 - Introduction to Object-Oriented Design and Development
Credits: 4
Principles of problem analysis and software design applied to the development cycle of a software system (i.e. from system requirements specification to design, implementation, and system test). Design and implementation using object-oriented principles, patterns, and tools. Experience in understanding and debugging software systems. Experience in working in groups. Prereq: CS 515.

CS 620 - Operating System Fundamentals
Credits: 4
Introduces operating system concepts and design. Process and memory management; scheduling; file systems; storage devices; inter-process communication. Prereq: CS 520.

CS 659 - Introduction to the Theory of Computation
Credits: 4

CS 671 - Programming Language Concepts and Features
Credits: 4
Explores the main features of modern, high-level, general purpose programming languages from the user (programmer) point of view. Provides students with an opportunity to use non-imperative programming paradigms, such as object-oriented, functional, and logical, and to learn how specific features of such languages can be used efficiently in solving programming problems. Prereq: CS 520.

CS 696 - Independent Study
Credits: 1-6
Individual projects developed and conducted under the supervision of a faculty member. Prereq: permission of faculty supervisor and department chairperson. May be repeated for credit.
CS 696W - Independent Study
Credits: 1-6
Individual projects developed and conducted under the supervision of a faculty member. Prereq: permission of faculty supervisor and department chairperson. May be repeated for credit. Writing intensive.
Attributes: Writing Intensive Course

CS 699 - Internship
Credits: 1
Provides the opportunity to apply academic experience in settings associated with future professional employment. A written proposal for the internship must be approved by the instructor. The proposal must specify what the student will learn from the internship, why the student is properly prepared for the internship and what supervision is available during the internship. A mid-semester report and final report are required. Prereq: permission. May be repeated for a maximum of 4 credits. Only open to Computer Science majors. Cr/F.

CS 712 - Compiler Design
Credits: 4
Formal languages and formal techniques for syntax analysis and parsing; organization of the compiler and its data structures; code generation. LL and LR parsing; automatic generation of scanners and parsers from high level descriptions. Implementation of features from imperative and object-oriented languages. Students required to design and implement a compiler for a simple language. Prereq: CS 520.

CS #720 - Operating System Programming
Credits: 4
Detailed discussion of operating system concepts and features. Practical examples and exercises that utilize advanced operating system features, including inter-process communication, synchronization, client-server communication, shared memory, threads, remote procedure calls, and device-level I/O. Discussion of POSIX 1003.1 Part I Standards. Prereq: CS 520 and CS 619.

CS 723 - Performance Evaluation of Computer Systems
Credits: 4
Introduces the main concepts, techniques, and tools needed to evaluate the performance of computer systems under various configurations and workloads. The techniques allow one to perform capacity planning based on quality of service requirements of users and workload characteristics. Course is mainly based on the use of analytic queuing network models of computers systems. The performance techniques are applied to study the performance of centralized, distributed, parallel, and client/server systems. The course also discusses performance measuring tools for operating systems such as Unix and Windows NT. Prereq: CS 620 and (MATH 539 or MATH 644).

CS 725 - Computer Networks
Credits: 4
Introduction to fundamental concepts of computer networks and exploration of widely-used networking technologies. Topics include principles of congestion and error control; network routing; local, wireless and access networks; application protocol design; and network programming. In-depth discussion of the Internet suite of protocols. Prereq: CS 520.

CS 730 - Introduction to Artificial Intelligence
Credits: 4
In-depth introduction to artificial intelligence, concentrating on aspects of intelligent problem-solving. Topics include situated agents, advanced search techniques, knowledge representation, logical reasoning techniques, reasoning under uncertainty, advanced planning and control, and learning. Prereq: CS 515.

CS 730W - Introduction to Artificial Intelligence
Credits: 4
In-depth introduction to artificial intelligence, concentrating on aspects of intelligent problem-solving. Topics include situated agents, advanced search techniques, knowledge representation, logical reasoning techniques, reasoning under uncertainty, advanced planning and control, and learning. Prereq: CS 671. Writing intensive.
Attributes: Writing Intensive Course

CS 733 - Mobile Robotics
Credits: 4
An introduction to the foundational theory and practices in mobile robotics. Topics include Kinematics of wheeled mobile robots, Sensors for mobile robots, Robot navigation and perception, Robot vision, Localization and mapping of mobile robots. Hands-on experience directed towards implementation with a real robot. Prereq: Programming course of Permission of instructor.

CS 735 - Introduction to Parallel and Distributed Programming
Credits: 4
Programming with multiple processes and threads on distributed and parallel computer systems. Introduces programming tools and techniques for building applications on such platforms. Course requirements consist primarily of programming assignments. Prereq: CS 520.

CS 745 - Formal Specifications and Verification of Software Systems
Credits: 4
Course focuses on the formal specification and verification of reactive systems, most notably concurrent and distributed systems. Topics relevant to these systems, such as non-determinism, safety and liveness properties, asynchronous communication or compositional reasoning, as discussed. We rely on a notation (TLA+, the Temporal Logic of Actions) and a support tool (TLC, the TLA+ Model Checker). Prereq: CS 520 and CS 659.

CS 750 - Machine Learning
Credits: 4
An introduction to fundamental concepts and common methods in machine learning. In addition to theoretical topics, the course involves hands-on experience in making predictions using synthetic and real-world datasets. Prereq: MATH 539 or MATH 644, and Programming course or Permission of instructor.

CS 753 - Information Retrieval
Credits: 4
Fundamental algorithms and techniques for text processing and text-based information retrieval systems. Topics include how to build an end-to-end information retrieval system, such as a Web search engine. Prereq: CS 515.

CS 757 - Mathematical Optimization for Applications
Credits: 4
This course introduces the foundations of mathematical optimization and reinforces them via applications. The content includes convex optimization, first and second-order methods, constrained problems, duality, linear and quadratic programming, as well as discrete and non-convex optimization. Applications will focus on machine learning methods but also include problems from engineering and operations research. Prereq: MATH 426; Programming proficiency in MATLAB, R, Java, C, Python, or equivalent.

University of New Hampshire

3
CS 758 - Algorithms
Credits: 4
An introduction to important concepts in the design and analysis of algorithms and data structures, including implementation, complexity analysis, and proofs of correctness. Prereq: CS 515 and CS 659.

CS 760W - Introduction to Human-Computer Interaction
Credits: 4
Human-computer interaction is a discipline concerned with the design, evaluation, and implementation of interactive computing systems for human use and with the study of major phenomena surrounding them. Prereq: CS 619 and CS 620. Writing intensive.
Attributes: Writing Intensive Course

CS #767 - Interactive Data Visualization
Credits: 4
Detailed discussion of how an understanding of human perception can help us design better interactive displays of data. Topics include color, space perception, object perception and interactive techniques. Students write interactive programs, give presentations and undertake a project designing and evaluating a novel display technique. Prereq: instructor’s permission.

CS 767W - Interactive Data Visualization
Credits: 4
Detailed discussion of how an understanding of human perception can help us design better interactive displays of data. Topics include color, space perception, object perception and interactive techniques. Students write interactive programs, give presentations and undertake a project designing and evaluating a novel display technique. Prereq: instructor’s permission. Writing intensive.
Attributes: Writing Intensive Course

CS 770 - Computer Graphics
Credits: 4
Input-output and representation of pictures from hardware and software points of view; interactive techniques and their applications; three-dimensional image synthesis techniques and their applications. Prereq: CS 671.

CS 771 - Web Programming Paradigms
Credits: 4
In this course you will learn languages to program the Web. Languages integrated into browsers, like JavaScript, and languages invoked on the server, like Ruby. You will also learn about frameworks, like Rails, and various techniques used to support the programming process. In addition, you will learn languages you will need to create, modify, and process Web documents. Although we will learn how to read and write in these languages, our primary goal will be an understanding of how the design of these multi-paradigm dynamic languages support the process of developing Web applications. Prereq: CS 671.

CS 775 - Database Systems
Credits: 4
Introduction to database management systems --- design, implementation, and usage --- with focus on the relational model. Data description, manipulation, and query language in the context of MySQL. Schema design and normalization; indexes, transaction processing. Web access of databases (PHP); overview of XML and noSQL systems. No credit if credit earned for IT 775. Prereq: CS 515 and MATH 531.

CS 780 - Topics
Credits: 1-4
Material not normally covered in regular course offerings. May be repeated for credit.

CS 791 - Senior Project I
Credits: 2
The principal goal of CS 791 is to develop precise functional specifications for the senior projects as well as a complete software design specification. The course will review and expand upon design concepts presented in previous courses, including UML, and CRC approach, and design patterns. Students apply these concepts to their own senior project. A significant component of the design includes specifications of the testing methodology to be used. Prereq: CS 619. Computer Science and CS: Bioinformatics majors only.

CS 792 - Senior Project II
Credits: 2
Continuation of CS 791: Senior Project I. Students complete the project by implementing their design. Students work in teams. Successful completion of this course fulfills the capstone experience requirement for Computer Science majors. Prereq: CS 791. Writing intensive.
Attributes: Writing Intensive Course

CS 799 - Thesis
Credits: 4
Students work under the direction of a faculty sponsor to plan and carry out independent research resulting in a written thesis. Required for honors-in-major. Prereq: Permission. May be repeated to 8 credits.
Attributes: Writing Intensive Course

https://ceps.unh.edu/directory/all