COMPUTER SCIENCE DEPARTMENT

http://www.uvm.edu/~cems/cs/

Computer Science (CS) is a vibrant subject with academic depth, enormous growth, and universal economic impact. Computers are now ubiquitous in society and influence the way we learn, the way we do science and business, and the way we interact with and understand our world.

Edsger Dijkstra (a renowned computer scientist, 1930-2002) is reputed to have said “Computer Science is no more about computers, than astronomy is about telescopes.” Rather, CS is aptly defined as the science of problem solving. CS requires a combination of logical thinking, creativity, problem decomposition, implementation, verification and validation, and teamwork. Computing Careers are extremely versatile, lucrative, and in tremendous and growing demand.

CURRICULA

At the undergraduate level, UVM Computer Science offers three bachelor’s degrees, an accelerated M.S. degree, a minor, and a non-degree Certificate in Computer Software:

- **BS CS**: The Bachelor of Science in Computer Science provides the most depth in computer science, complemented by breadth in math, science, humanities, and social sciences. The BS CS is offered through the College of Engineering & Mathematical Sciences.

- **BS CSIS**: The Bachelor of Science, major in Computer Science and Information Systems, is an interdisciplinary degree that combines computer science with business, offering a competitive combination of skills and knowledge. The BS CSIS is offered through the College of Engineering and Mathematical Sciences, in cooperation with the School of Business Administration.

- **BA CS**: The Bachelor of Arts, major in Computer Science, provides a computer science major in the context of a liberal education, and has sufficient flexibility to facilitate a double major in another field such as mathematics, biology, music, etc. The BA is offered through the College of Arts and Sciences, and information for this program can be found under the Arts and Sciences portion of the Undergraduate Catalog Website.

- **AMP**: The Accelerated Masters Program is open to academically strong CS juniors in any of our 3 undergraduate majors. The AMP allows students to apply two CS upper division courses towards both a bachelor’s and master’s degree, enabling completion of the Master of Science in CS in as little as one additional year beyond their Bachelor’s degree. Information on the AMP can be found under the Graduate Catalog Website.

- **CS minor**: The minor in Computer Science is a flexible 6-course program, which is a great complement to virtually any other UVM major and adds marketable skills.

- **Certificate in CS**: A non-degree Certificate in Computer Software is a flexible 5-course program offered jointly with the Division of Continuing Education. It can be used to obtain career skills or to make up pre-requisites for the MS program in CS. Information about this program can be found on the Continuing Education Website.

UVM CS courses provide a mixture of lecture-based and hands-on experiential learning exercises. Our curricula provide a solid foundation in both applied and theoretical aspects of computing, preparing students for future careers and/or graduate study in computing. Many of our students complete paid internships during their summers.

ACADEMIC STANDARDS

In order to continue as a major in the Department of Computer Science in CEMS, a student must achieve a 2.00 cumulative grade-point average at the end of the semester in which 60 cumulative credits have been attempted. No more than three repeated course enrollments are allowed during this 60-credit period. In the case of transfer students, applicable transfer credits will be included in determining the 60 credits, but grades in these courses will not be included in the grade-point average.

Students who receive a cumulative or semester grade-point average of less than 2.00 will be placed on trial. Students who have failed half their course credits for any semester, or who have had two successive semester averages below 2.00, or three successive semesters in which their cumulative grade-point average falls below 2.00, are eligible for dismissal.

To receive a degree, students must have a minimum cumulative average of 2.00. Students must complete 30 of the last 45 hours of credit in residence at UVM as matriculated students in the College of Engineering and Mathematical Sciences.

No more than three grades of D+, D, or D- in computer science courses numbered CS 124 and higher may be applied to the Bachelor of Science in Computer Science. No more than three grades of D+, D, or D- in computer science courses numbered CS 124 and higher or BSAD courses numbered 100 and higher may be applied to the Bachelor of Science with a major in Computer Science & Information Systems.

MAJORS

**COMPUTER SCIENCE MAJORS**

- Computer Science B.S.CS.
- Computer Science and Information Systems B.S.

MINORS

**COMPUTER SCIENCE MINOR**

Computer Science
Courses

CS 002. MS Office: Beyond the Basics. 0 or 3 Credits.
Word documents looking dull? Excel charts lacking something? PowerPoint slides fizzling? All this and more is covered. Learn more than just the basics.

CS 005. Introductory Special Topics. 0-3 Credits.
Prerequisite: Instructor permission. Hours variable. May not be taken for credit after any Computer Science course numbered CS 016 or higher.

CS 008. Intro to Web Site Development. 0 or 3 Credits.
Provides a strong foundation in HTML, CSS, images, beginning web programming, and web design so that the student can create a complete functional web site.

CS 014. Visual Basic Programming. 3 Credits.
Introduction to Microsoft’s rapid development environment. Create playful and relevant Windows applications.

CS 019. Introduction to Programming. 0 or 3 Credits.
A gentle, graphical introduction to computer programming. Pre/co-requisite: No credit after CS 021 or higher.

CS 020. Programming for Engineers. 0 or 3 Credits.
Introduction to computer programming principles using MATLAB, with applications chosen from civil, electrical, environmental, and mechanical engineering. Co-requisite: MATH 021. Cross-listed with: ENGR 020. Credit not given for both CS 016 and CS 020/ENGR 020.

CS 021. Computer Programming I. 0 or 3 Credits.
Introduction to algorithmic problem solving. Designed to provide a foundation for further studies in computer science. Prerequisites: MATH 010 or a strong background in secondary school algebra and trigonometry.

CS 031. C Programming. 1-3 Credits.
Introduction to C programming for those already familiar with another programming language. Variable types, pointers, memory allocation, input/output, math, time, and other library calls. Prerequisites: One of CS 016, CS 020, CS 021 or equivalent.

CS 032. Puzzles, Games & Algorithms. 0 or 3 Credits.
Introductory computer science through exploration and analysis of mathematical puzzles and games, and the algorithms that handle them.

CS 042. Dynamic Data on the Web. 3 Credits.
Data is everywhere; Learn to collect, organize, and classify it. Students will design and create tables, queries and reports on the web using introductory programming.

CS 050. Seminar for New CS Majors. 1 Credit.
A fun and accessible breadth-first introduction to the CS community and curricula at UVM. CS faculty serve as guest lecturers to introduce new CS majors to selected topics covered in upper division UVM CS electives. Prerequisites: Computer Science or Computer Science & Information Systems majors who have not yet completed CS 110. Co-requisite: CS 021 or CS 110.

CS 064. Discrete Structures. 3 Credits.
Introduction to analytic and formal methods of computer science with practical examples, including analysis or data structures, recursion relations, proof methods, and logic programming. Credit not given for more than one of CS 064, MATH 052 or MATH 054. Co-requisites: One semester of programming, MATH 020 or MATH 022.

CS 095. Special Topics. 0-18 Credits.
See Schedule of Courses for specific titles. Prerequisite: Instructor permission.

CS 100. Object-Oriented Programming. 3 Credits.
Object-oriented software analysis, design, and programming using a modern object-oriented programming environment. Topics include encapsulation, information hiding, inheritance, and polymorphism. Prerequisite: CS 026 or CS 110.

CS 110. Intermediate Programming. 0 or 4 Credits.
Intermediate programming concepts including common data structures, algorithms, style, design, documentation, testing and debugging techniques, and an introduction to object-oriented programming. Prerequisites: One of CS 016, CS 020, CS 021 or equivalent.

CS 121. Computer Organization. 0 or 3 Credits.
Introduction to computer system organization including performance, assembly language, machine-level data representation, arithmetic for computers, processor datapath control, memory, and input/output. Prerequisite: CS 026 or CS 110. No credit for both CS 101 and CS 121.

CS 123. Programming Languages. 3 Credits.
Systematic treatment of principles underlying the features and implementation of programming languages. Contrast of traditional procedural languages and at least one nontraditional language. Prerequisites: CS 026 or CS 110; CS 064 or MATH 052 or MATH 054. No credit for both CS 103 and CS 123.

CS 124. Data Structures & Algorithms. 3 Credits.
Design and implementation of linear structures, trees and graphs. Examples of common algorithmic paradigms. Theoretical and empirical complexity analysis. Sorting, searching, and basic graph algorithms. Prerequisites: CS 026 or CS 110, CS 064 or MATH 052 or MATH 054. No credit for both CS 104 and CS 124.
CS 125. Computability and Complexity. 3 Credits.
Formal languages and expressiveness. Turing completeness and Church’s Thesis. Decidability and tractability. Complexity classes and theory of NP completeness. Prerequisites: CS 064 or MATH 052. Co-requisite: CS 124.

CS 128. Probability Models & Inference. 3 Credits.
Introduction to probability and statistics with computer science applications: probability spaces, discrete and continuous random variables, distributions, conditional probability, Markov chains, statistical estimation and regression. Prerequisites: CS 064 or MATH 052.

CS 142. Advanced Web Design. 3 Credits.
Advanced web site design, including structure, architecture, compliance, CSS, usability, etc., to help create a pleasing user experience. Prerequisite: CS 008.

CS 148. Database Design for the Web. 3 Credits.
Design and implementation of a relational database model using SQL and PHP. Typical project includes creation of ecommerce shopping site. Prerequisite: CS 008.

CS 189. CS for Geospatial Technologies. 3 Credits.
Introductory course providing hands-on experience with activities involving programming languages, platforms, and technologies in use by the GIS programmer/developer. Prerequisite: One course in GIS (CE 010, GEOG 081, or NR 143) or one in computer programming.

CS 192. Independent Service & Teaching. 1-3 Credits.
Independently designed project or pedagogical experience that benefits the University or the Community under the direction of a CS faculty member. Requires final presentation. Pre/co-requisite: Department permission.

CS 195. Special Topics. 1-9 Credits.
See Schedule of Courses for specific titles. Prerequisite: Instructor permission.

CS 201. Operating Systems. 0 or 3 Credits.
Supervisory and control software for multiprogrammed computer systems. Processes synchronization, interprocess communication, scheduling, memory management, resource allocation, performance evaluation, object-oriented systems, case studies. Prerequisites: CS 101 or CS 121; CS 104 or CS 124.

CS 204. Database Systems. 3 Credits.
Techniques for processing very large collections of data. Secondary storage. Database design and management. Query languages and optimization. Database recovery. Prerequisite: CS 104 or CS 124.

CS 205. Software Engineering. 3 Credits.
Treatment of software engineering problems and principles, including documentation, information hiding, and module interface specification syntax and semantics. Requires participation in a team project. Students who receive credit for CS 205 may not receive credit for CS 208 or CS 209. Prerequisite: CS 104 or CS 124. Cross-listed with: CSYS 205.

CS 206. Evolutionary Robotics. 3 Credits.
Exploration of the automated design of autonomous machines using evolutionary algorithms. Coursework involves reading of research papers, programming assignments and a final project. Prerequisites: Junior standing and programming experience, or Instructor permission.

CS 222. Computer Architecture. 3 Credits.
Architecture of computing systems. Control unit logic, input/output processors and devices, asynchronous processing, concurrency, parallelism, and memory hierarchies. Prerequisite: CS 101 or CS 121.

CS 224. Algorithm Design & Analysis. 3 Credits.
Comprehensive study of algorithms including greedy algorithms, divide and conquer, dynamic programming, graph algorithms and network flow. Computational intractability. Approximation, local search and randomization. Prerequisite: CS 124. CS 125 and one course in probability (e.g. STAT 143, STAT 151 or CS 128) are recommended.

CS 228. Human-Computer Interaction. 3 Credits.
The design, implementation and evaluation of user interfaces for computers and other complex, electronic equipment. Includes a significant project. Pre/co-requisites: Programming experience and Junior standing or Instructor permission.

CS 231. Programming for Bioinformatics. 3 Credits.
Introductory course on computing (including scripting, database, and statistical analysis) for developing bioinformatics applications. Particular emphasis is given to comparative genomics and systems biology scenarios. Prerequisites: STAT 151, STAT 153 or Instructor permission. Cross-listed with: MMG 231.

CS 232. Methods in Bioinformatics. 3 Credits.
This course provides a methodological survey of bioinformatics. Particular emphasis is given to algorithms associated with sequence analysis, comparative genomics, structural biology, and systems biology. Prerequisites: STAT 151, STAT 153, or Instructor permission. Cross-listed with: MMG 232.

CS 243. Theory of Computation. 3 Credits.
Reducibility and decidability, recursion theory, time and space complexity, P, NP, NP-completeness, PSPACE, PSPACE-completeness, L and NL, advanced topics in computability and complexity. Prerequisites: CS 124, CS 125.

CS 251. Artificial Intelligence. 3 Credits.
Introduction to methods for realizing intelligent behavior in computers. Knowledge representation, planning, and learning. Selected applications such as natural language understanding and vision. Prerequisites: CS 103 or CS 123; CS 104 or CS 124; STAT 153 or equivalent. Cross-listed with: CSYS 251.

CS 254. Machine Learning. 3 Credits.
Introduction to machine learning, including supervised and unsupervised learning algorithms, reinforcement learning, and computational learning theory. Prerequisites: CS 128, STAT 151 or STAT 153 or equivalent, MATH 121, MATH 124.
CS 256. Neural Computation. 3 Credits.
Introduction to artificial neural networks, their computational capabilities and limitations, and the algorithms used to train them. Statistical capacity, convergence theorems, backpropagation, reinforcement learning, generalization. Prerequisites: MATH 124 or MATH 271; STAT 153 or equivalent; computer programming. Cross-listed with: STAT 256, CSYS 256.

CS 260. Parallel Computing. 3 Credits.
Taxonomy of parallel computers, basic concepts for parallel computing, effectiveness and scalability, parallel algorithms for variety of problems, distributed memory and shared memory paradigms. Prerequisite: CS 104 or CS 124, or Instructor permission.

CS 265. Computer Networks. 3 Credits.
Introduction to the theoretical and pragmatic principles and practices of computer networking. Topics include: local area networks; the Internet; network and world-wide-web application programming. Prerequisites: CS 026 or CS 110, CS 101 or CS 121, and STAT 153 or equivalent.

CS 266. Network Security&Cryptography. 3 Credits.

CS 274. Computer Graphics. 3 Credits.
Graphical representation of two- and three-dimensional objects on color raster displays. Line generation, region filling, geometric transformations, hidden line and surface removal, rendering techniques. Prerequisite: CS 104 or CS 124, MATH 124 or MATH 271, recommended.

CS 275. Mobile Apps & Embedded Devices. 3 Credits.
A projects-based course focused on applications development on wireless and embedded platforms, including iOS, Arduino, and Linux-based devices. Emphasis on C programming and cyber-physical systems software. Prerequisite: CS 124. Pre/Co-requisites: CS 148 or CS 204 (recommended but not required).

CS 276. Integrative Computing. 3 Credits.
Integrative computing principles and practices: Abstraction via APIs, distributed systems orchestration, security, application design and implementation. Team projects for mobile and other networked, embedded devices. Prerequisites: Senior standing in Computer Science or Instructor permission.

CS 283. Undergraduate Honors Thesis. 3 Credits.
See description of Honors Thesis Program in the College of EM section of this catalog.

CS 284. Undergraduate Honors Thesis. 3 Credits.
See description of Honors Thesis Program in the College of EM section of this catalog.

CS 292. Senior Seminar. 1 Credit.
Oral presentations that pertain to the ethical practice of computer science in government, industry, and academia. Topics may include computer security, copyright, and patent law. Prerequisite: Senior standing in Computer Science.

CS 294. Independent Readings&Research. 1-6 Credits.
Independent readings and investigation under the direction of faculty member. Prerequisite: Department permission.

CS 295. Special Topic:Computer Science. 1-18 Credits.
See Schedule of Courses for specific titles. Subject will vary from year to year. May be repeated for credit.

CS 296. Special Topic:Computer Science. 1-12 Credits.
See Schedule of Courses for specific titles. Subject will vary from year to year. May be repeated for credit.