

COMPUTER SCIENCE

Department Overview

Computer Science is an academic discipline that is fascinating to study in its own right but also has very wide-ranging applications throughout the modern world. Our faculty are all skilled and dedicated teachers as well as active scholars; we strive to make each course we offer engaging and challenging. Our graduates go on to a great variety of careers in such areas as theoretical computer science, the computer science industry, teaching at various levels, and many more.

Chair of the Computer Science Department: Christine Reilly

Associate Chair of the Computer Science Department: Thomas O'Connell

Associate Professors: Michael Eckmann, Thomas O'Connell, Aarathi Prasad, Christine Reilly

Lecturers: Nelson Dellis, David Read

Linux Systems Administrator: Lisa Schermerhorn

Administrative Assistant: Kim Newsom

Computer Science B.A.

Effective for Students Who Entered Skidmore in Fall 2020 and Beyond

Students majoring in computer science fulfill the departmental requirements by completing the following:

Code	Title	Hours
Required Computer Science Courses		
CS 209	Data Structures and Mathematical Foundations	4
CS 226	Software Design	4
CS 230	Programming Languages	4
CS 305	Design and Analysis of Algorithms ¹	4
CS 306	Computability, Complexity, and Heuristics	4
CS 318	Introduction to Computer Organization	4
Required Mathematics Courses		
MA 200	Linear Algebra	4
Select one of the following:		3-6
MA 111	Calculus I	
MA 108 & MA 109	Calculus with Algebra I and Calculus with Algebra II	
MA 113	Calculus II	
MA 211	Calculus III	
Electives		
Select one of the following: ²		3
One CS course at the 200 level or above ³		
MA 302	Graph Theory	
MA 305	Introduction to Probability	
MA 316	Numerical Algorithms	
Select two CS courses at the 300 level ^{4,5}		6
Total Hours		40-43

¹ In CS 305 Design and Analysis of Algorithms, students will fulfill the writing requirement in the major by acquiring the skills necessary

to clearly describe algorithmic processes and argue about their properties.

- ² With permission of the department, a student may instead take a course in another discipline that has substantial computer science content.
- ³ Excluding CS 275 Introduction to Research in Computer Science, CS 275H Introduction to Research in Computer Science, CS 371 Independent Study in Computer Science, CS 381 Senior Thesis, and CS 382 Senior Thesis.
- ⁴ Excluding CS 371 Independent Study in Computer Science, CS 381 Senior Thesis, and CS 382 Senior Thesis.
- ⁵ Includes IL 305

No more than 8 credit hours of S/U can be used toward the major.

The following courses must be taken for a letter grade: CS 305 Design and Analysis of Algorithms, CS 306 Computability, Complexity, and Heuristics, and CS 318 Introduction to Computer Organization.

Effective for Students Who Entered Skidmore Prior to Fall 2020

Students majoring in computer science fulfill the departmental requirements by completing the following:

Code	Title	Hours
Required Computer Science Courses		
CS 206	Introduction to Computer Science II	4
CS 230	Programming Languages	4
CS 305	Design and Analysis of Algorithms	4
CS 306	Computability, Complexity, and Heuristics	4
CS 318	Introduction to Computer Organization	4
Required Mathematics Courses		
MA 200	Linear Algebra	4
MA 215	Introduction to Mathematical Reasoning and Proof ¹	4
Electives		
Select one of the following: ²		3
MA 302	Graph Theory	
MA 305	Introduction to Probability	
MA 316	Numerical Algorithms	
One CS course at the 200 level or above ³		
Select two CS courses at the 300 level ⁴		6
Total Hours		37

¹ In MA 215 Introduction to Mathematical Reasoning and Proof, students will acquire writing skills that are necessary to work on advanced material in mathematics and will fulfill the writing requirement in the major.

² With permission of the department, a student may instead take a course in another discipline that has substantial computer science content.

³ Excluding CS 275 Introduction to Research in Computer Science, CS 275H Introduction to Research in Computer Science, CS 371 Independent Study in Computer Science, CS 381 Senior Thesis, and CS 382 Senior Thesis.

⁴ Excluding CS 371 Independent Study in Computer Science, CS 381 Senior Thesis, and CS 382 Senior Thesis.

No more than 8 credit hours of S/U can be used toward the major.

The following courses must be taken for a letter grade: CS 305 Design and Analysis of Algorithms, CS 306 Computability, Complexity, and Heuristics, CS 318 Introduction to Computer Organization.

Computer Science Minor

Effective for Students Who Entered Skidmore in Fall 2020 and Beyond

Code	Title	Hours
CS 209	Data Structures and Mathematical Foundations	4
Select one of the following:		4
CS 305	Design and Analysis of Algorithms	
CS 318	Introduction to Computer Organization	
CS 230	Programming Languages	
Select three additional courses of the following: any two or three CS courses (at most one of which may be at the 100 level)		9-12
MA 316	Numerical Algorithms	
Total Hours		17-20

Effective for Students Who Entered Skidmore Prior to Fall 2020

Students minoring in computer science fulfill the departmental requirements by completing the following:

Code	Title	Hours
CS 206	Introduction to Computer Science II	4
Select two of the following:		8
CS 305	Design and Analysis of Algorithms	
CS 318	Introduction to Computer Organization	
CS 230	Programming Languages	
Select two additional courses of the following:		6-8
MA 215	Introduction to Mathematical Reasoning and Proof	
MA 302	Graph Theory	
MA 316	Numerical Algorithms	
any CS courses (one of which may be at the 100 level)		
Total Hours		18-20

Honors

Students wishing to qualify for departmental honors in the computer science major must:

1. Complete all departmental requirements for the computer science major and have a GPA of 3.5 or higher for all course work (MC and CS) taken in the department, together with a course in another discipline that has substantial computer science content if taken as an elective with permission of the department;
2. Have a GPA of 3.0 for all course work taken at Skidmore;
3. File with the department, by the end of the official add-drop period of the fall semester of the senior year, a declaration of intention to qualify for honors; and
4. Submit an honors thesis to be read by a review committee, and give an oral presentation of the thesis to the department. The review committee will evaluate the thesis to determine if it is of the

exceptional quality that merits honors; the committee will submit its recommendation to the department for final adjudication.

Additional Guidance Regarding the Honors Thesis

An honors thesis may take a variety of forms. The following guidelines provide some clarity regarding the department's expectations. In all cases, further discussion with your thesis advisor will be necessary before formalizing your research plan.

A computer science honors thesis will:

- Require at least one full academic year of applied effort to complete;
- Involve significant review of existing research in the area of interest. The thesis will include an extensive literature review section highlighting relevant concepts, frameworks, and theories; and
- Result in a creative contribution. The contribution might be an implementation and validation of existing work or an identification of a gap in existing literature, along with a validation of your proposed solution. The creative contribution must confirm the student's grasp of the subject matter while demonstrating independent thinking.

Note: All CS and MC courses have satisfaction of QR1 as a prerequisite.

CS 102A - Computing in Context

Credits: 4

A set of courses exploring interesting applications of computing in a variety of disciplines. These courses are primarily intended for students who wish to satisfy the QR requirement and enhance their abilities to apply computing to the solution of quantitative problems.

Prerequisites: Students who have not already completed CS 206 may enroll. Courses are offered periodically depending on faculty availability.

Note(s): Fulfills QR2 requirement.

CS 102C - Computing in Context

Credits: 3

A set of courses exploring interesting applications of computing in a variety of disciplines. These courses are primarily intended for students who wish to satisfy the QR requirement and enhance their abilities to apply computing to the solution of quantitative problems. Courses are offered periodically depending on faculty availability.

Note(s): Fulfills QR2 requirement.

CS 102D - Computing in Context

Credits: 4

A set of courses exploring interesting applications of computing in a variety of disciplines. These courses are primarily intended for students who wish to satisfy the QR requirement and enhance their abilities to apply computing to the solution of quantitative problems. Courses are offered periodically depending on faculty availability.

Note(s): Fulfills QR2 requirement.

CS 103 - Structured Prog Basic

Credits: 3

CS 106 - Introduction to Computer Science I

Credits: 4

An introduction to the principles of design, implementation, and testing of computer programs. The course covers language features such as control structures, classes, file I/O, and basic data structures including arrays. Other topics include recursion and fundamental algorithms, such as elementary searching and sorting algorithms. May not be taken for credit by students who have taken or are currently taking CS 107, CS 206, CS 209 or CS 226.

Prerequisites: Placement in AQR or completion of a FQR course or MA 108 or PH207.

Note(s): May not be taken for credit by students who have taken or are currently taking CS 107, CS 206, CS 209 or CS 226. Fulfills QR2 requirement.

CS 107 - An Introduction to Computer Science with Animations and 2D Games

Credits: 4

An introduction to computer science. Students will develop simple 2D games, animations and simulations using the Java programming language and Processing, a Java package designed to make programming accessible to visual artists. The course introduces the fundamental techniques of computer programming, software design, and algorithm development.

Note(s): May not be taken for credit by students who have taken or are currently taking CS 106, CS 206, CS 209 or CS 226.

CS 206 - Introduction to Computer Science II

Credits: 4

Fundamentals of software development and algorithm design. Topics include recursion, data structures, analysis of algorithms, and program verification.

Prerequisites: CS 106 or CS 107 or permission of instructor.

CS 209 - Data Structures and Mathematical Foundations

Credits: 4

An introduction to data structures and the mathematical foundations of algorithms. Properly organizing the data within a software application is critical to the efficiency of the application's execution as well as the ease with which the application is developed. This course will introduce the fundamental data structures used in computer science, including linked lists, stacks, queues, priority queues, search trees, and hash tables. In addition, the course will introduce foundational discrete mathematical concepts used in the design and analysis of algorithms, including mathematical induction, asymptotic analysis and recurrence relations.

Prerequisites: CS 106 or CS 107 or permission of instructor.

CS 225 - Applied Data Science

Credits: 4

An introduction to Knowledge Discovery and Data Mining (KDD), also known as Machine Learning. Students will gain hands-on experience with a variety of concepts and tools associated with the field of KDD. Students will also study and apply KDD processes, tools, and techniques, working with small and large data sets representing various business domains and problem types. The class will cover topics such as common database interactions (relational, document, and graph databases), data set creation, data quality analysis, dimension reduction, variance and bias in data, classifier types, and model assessment approaches.

Prerequisites: MA 104 or MA 204 or a similar statistics course.

CS 226 - Software Design

Credits: 4

An introduction to building complex software. Students will explore object-oriented (OO) programming and design patterns in depth. We will also focus on developing skills for testing and debugging of software. Finally, we will utilize a set of frameworks related to software development and testing.

Prerequisites: CS 106 or CS 107.

CS 230 - Programming Languages

Credits: 4

An introduction to different programming language paradigms: functional, logic, and object-oriented programming. Students will also study language concepts such as regular expressions, syntax grammars and semantics. Specific topics may include Perl, Scheme, Java, C++, and Prolog.

Prerequisites: CS 206 or CS 226 and CS 209.

CS 275 - Introduction to Research in Computer Science

Credits: 1

An introductory exploration of research in computer science. The students, in collaboration with a faculty mentor, will participate in a research project in a particular area of computer science. The research projects may, for example, include designing new algorithms for computational problems, surveying the research literature, implementing existing algorithms from the research literature, or performing computational experiments.

Prerequisites: Permission of instructor.

Note(s): Students may enroll for CS275 four times in their careers, but may take no more than two in any given semester. Does not count toward the CS major. Must be taken S/U.

CS 275H - Introduction to Research in Computer Science

Credits: 1

An introductory exploration of research in computer science. The students, in collaboration with a faculty mentor, will participate in a research project in a particular area of computer science. The research projects may, for example, include designing new algorithms for computational problems, surveying the research literature, implementing existing algorithms from the research literature, or performing computational experiments.

Prerequisites: Permission of instructor.

Note(s): Students may only take four CS 275H courses in their careers and may take no more than two in any given semester. If two are taken in a single semester, each must be a different section. CS 275H may not be counted toward the CS major. Must be taken S/U.

CS 276 - Selected Topics in Computer Science

Credits: 1-4

Topics that complement the established lower level course offerings in computer science will be selected. May be repeated for credit.

Prerequisites: Permission of instructor.

CS 277 - Topics in Computer Science with Lab

Credits: 4

Topics that complement the established lower level course offerings in computer science will be selected.

Prerequisites: Permission of instructor.

Note(s): May be repeated for credit.

CS 305 - Design and Analysis of Algorithms

Credits: 4

A study of techniques used to design algorithms for complex computational problems that are efficient in terms of time and memory required during execution. Students will also learn the techniques used to evaluate an algorithm's efficiency. Topics include advanced sorting techniques, advanced data structures, dynamic programming, greedy algorithms, amortized analysis, graph algorithms, network flow algorithms, and linear programming.

Prerequisites: CS206 and MA215, or CS209; and a calculus course.

CS 306 - Computability, Complexity, and Heuristics

Credits: 4

Using abstract machine models to measure the complexity of computational problems and to explore the limits of computational power. In addition, students will investigate strategies to deal with intractable computational problems when they arise. Topics include finite automata, Turing machines, undecidability, NP-completeness, approximation algorithms, heuristic algorithms, and experimental algorithmics.

Prerequisites: CS 305.

CS 316 - Foundations of Machine Learning

Credits: 4

An introduction to the mathematical foundations of machine learning. Students will investigate the mathematics underlying much of machine learning and explore some of the major techniques used in the field. Topics may include gradient descent, perceptrons, support vector machines, kernel methods, backpropagation, and principal components analysis. Three hours of lecture and two hours of lab each week. In the labs, students will apply machine learning techniques to empirical data and quantitatively evaluate and compare the effectiveness of the techniques.

CS 318 - Introduction to Computer Organization

Credits: 4

An introduction to multilevel machines, including basic components of a computer, digital circuits, Boolean algebra, microprogramming, machine and assembly languages, and operating systems.

Prerequisites: CS 206 or CS 226.

Note(s): Does not count toward the mathematics major.

CS 322 - Artificial Intelligence

Credits: 4

An introduction to the field of artificial intelligence. The course covers the main techniques used to develop computer programs to solve problems that we normally think of as requiring intelligence. Topics include search, games, knowledge representation, logical reasoning systems, and machine learning.

Prerequisites: MA 111, or both MA 108 MA 109, or MA 113; and CS 305; and CS 206 or CS 226.

Note(s): Fulfills Senior Coda requirement.

CS 325 - Computer Graphics

Credits: 4

Computer graphics involves using computers to generate images, as opposed to generating images using cameras. Computer graphics images typically try to mimic reality. In this course, students will explore the necessary background for further study in computer graphics. Students will explore the basics of human vision that influence the way computers generate images; projections from three-dimensional space to two-dimensional space; various models of real-world entities such as lighting, surface reflectance, and color; and classic algorithms in computer graphics that students will implement and with which they will have the opportunity to experiment.

Prerequisites: CS 226 and CS 209, or CS 206; and MA 200.

CS 326 - Software Engineering

Credits: 4

An exploration of software engineering and organization principles vital to building large enterprise-scale applications. Leveraging the lessons-learned from readings and using management tools, the class will form a software development team focused on designing, developing, and deploying a multi-tier application. Includes significant reading and programming effort.

Prerequisites: CS 226 and CS 209; or CS 206.

Note(s): Fulfills Senior Coda requirement.

CS 327 - Computer Networks

Credits: 4

An introduction to the fundamental concepts of computer networks. Students will learn how the Internet works. Students will explore real data as it passes through a network; learn how data is interpreted and forwarded from the source to the destination address; implement services to send, transport, and receive data correctly; devise analytical techniques to better understand data collected from real network users; and gather information about known network security attacks.

Prerequisites: (CS 206 or CS 209) and CS 318.

CS 328 - Mobile Computing

Credits: 4

An introduction to the design, implementation and publishing of mobile applications. Students will learn how to take their innovative ideas from conception to a working prototype mobile application through a series of rigorous hands-on programming assignments and group projects.

Prerequisites: CS 226 and CS 209; or CS 206.

Note(s): Fulfills Senior Coda requirement.

CS 329 - Operating Systems

Credits: 4

An introduction to operating systems: software that safely and fairly shares computer hardware resources among multiple, simultaneously running applications. Topics covered include computer processor (CPU) sharing and scheduling, memory virtualization and management, and file systems. Students will complete programming projects to implement concepts in a real operating system.

Prerequisites: (CS 206 or CS 209) and CS 318.

CS 331 - Computer Vision

Credits: 4

Computer Vision (CV) trains machines to understand images and video. Classifying and detecting/recognizing objects in images are common problems that CV tackles. Machine Learning techniques, specifically Convolutional Neural Networks (CNNs), are often used. Students will learn digital image processing techniques that form the basis of many vision algorithms, as well as the concepts behind CNNs for classification.

Prerequisites: CS209 and MA200 and (MA111 or MA113 or (MA108 and MA109)).

Note(s): Fulfills Senior Coda requirement.

CS 355 - Database Systems

Credits: 4

An introduction to database systems: software that stores, manages, and provides efficient access to a collection of structured information. Topics covered include the relational model, the structured query language (SQL), and algorithms and data structures for efficiently searching and managing data that is larger than main memory. Students will complete a project using a relational database to store and query data related to an application of their choice.

Prerequisites: CS 206 or CS 209.

Note(s): Fulfills Senior Coda requirement.

CS 371 - Independent Study in Computer Science

Credits: 1-4

Special study in computing outside of the regular departmental offerings.

Prerequisites: Consent of department.

CS 376 - Advanced Topics In Computer Science

Credits: 3,4

Advanced topics that complement the established course offerings in computer science will be selected.

Prerequisites: Permission of instructor.

Note(s): May be repeated for credit.

CS 381 - Senior Research

Credits: 3

Optional for computer science majors. Recommended for those working toward professional careers or graduate study in computer science, and those seeking to satisfy the criteria for departmental honors.

Note(s): Fulfills Senior Coda requirement.

CS 382 - Senior Research

Credits: 3

Optional for computer science majors. Recommended for those working toward professional careers or graduate study in computer science, and those seeking to satisfy the criteria for departmental honors.

Note(s): Fulfills Senior Coda requirement.

CS 399 - Internship in Computer Science

Credits: 1-9

Professional experience at an advanced level for juniors and seniors with substantial academic experience in computer science and mathematics. With faculty sponsorship and departmental approval, students may extend their educational experience in computer science, software engineering, or applied mathematics.

Prerequisites: CS 206 or CS 226 or CS 209 and one additional course in mathematics or computer science at the 200 level or above and permission of the department.

Note(s): Fulfills Senior Coda requirement. This course may not be used to satisfy the requirements of any major or minor in the department. Not for liberal arts credit. Must be taken S/U.