

Introductory Computer Science Courses

The Department of Computer Science offers a range of introductory courses to accommodate students of varying backgrounds and interests.

CPSC 079a *Digital Photorealism.*

Examination of basic methods used to define shapes, materials, and lighting when creating computer-generated images. Topics include mathematical models for shape, texture models, and lighting techniques. Principles are applied through use of modeling/rendering software. The term project will be the production of a short animated video with rich visual effects. *Proficiency in high-school-level mathematics is assumed. No previous experience with computers necessary.* (Preregistration required; open only to freshmen.)

CPSC 112a or b *Introduction to Programming.*

An introductory course designed to teach students majoring in any subject how to program computers. The language taught is either JAVA or C#. The focus is on the development of programming skills, problem-solving methods, and selected applications. Topics include data types, control structures, basic algorithms, object-oriented programming, graphical user interfaces, and some advanced programming concepts. *No previous experience with computers necessary.*

CPSC 150a / HUMS 407a *Computer Science and the Modern Intellectual Agenda.*

An introduction to the basic ideas of computer science (computability, algorithm, virtual machine, symbol processing system) and of several ongoing relationships between computer science and other fields, including philosophy of mind, classical cognitivism, connectionism, and artificial life. *No previous experience with computers necessary.* (Satisfies the WR and HU requirements. Enrollment limited to 25.)

CPSC 151b / HUMS 408b *The Graphical User Interface: DOS to Windows to What?.*

(Not taught in 2011–2012)

The role of Graphical User Interfaces (such as the Desktop, with its overlapping windows, icons, menus and pointer device—as embodied in Mac OS, Microsoft Windows etc), on standard platforms such as desktop PCs, laptops, small-screen devices etc. Why did GUIs develop in the way they did? Why have they evolved so little since the Desktop of the 1970s? How will changing hardware and user requirements reshape them in the future? *Prerequisite: Have used a desktop or laptop computer.* (Satisfies the WR requirement. Enrollment limited to 25.)

CPSC 163b / ARCG 163b / HUMS 338b / NELC 163b *From Pictograph to Pixel: Changing Ways of Human Communication.*

An exploration of the five pivotal stages in the development of human communication: pictographic and syllabic ways of writing; the consonantal or phonetic alphabet; the invention of paper; movable type; and electronic/digital media and the Internet. These technologies are considered for their innovative features, new capabilities, social and ideological implications, and the instrumental parts they played in contemporary periods of change. (Not taught every year.)

CPSC 183a *Introduction to Law and Technology.*

An exploration of the myriad of ways in which law and technology intersect, with a special focus on the role of cyberspace. The course lays out a basic framework for the many issues that arise in our modern legal and technological contexts. It covers topics such as digital copyright, free speech, privacy and anonymity, information security, innovation, online communities, the impact of technology on society, and emerging trends. *No technical knowledge or previous coursework required.* (Satisfies the SO requirement.)

CPSC 185b *Control, Privacy, and Technology.*

A case-law and policy intensive class that explores how various legal doctrines have evolved with and around technological development. Topics include criminal law, privacy, search and seizure, digital rights, and the implications of technologically-permitted (and enforced) methods of control on the law. *After CPSC 180a or b or 183a.* (Satisfies the WR and SO requirements. Enrollment limited to 24.)

CPSC 201a or b *Introduction to Computer Science.*

An introduction to the concepts, techniques, and applications of computer science for potential majors. Topics include computer systems (the design of computers and their languages); theoretical foundations of computing (computability, complexity, algorithm design); and artificial intelligence (the organization of knowledge and its representation for efficient search). Examples stress the importance of different problem-solving methods. *After CPSC 112a or b or equivalent.*

CPSC 202a *Mathematical Tools for Computer Science.*

Introduction to formal methods for reasoning and to mathematical techniques basic to computer science. Topics include propositional logic, discrete mathematics, and linear algebra. Emphasis on applications to computer science: recurrences, sorting, graph traversal, Gaussian elimination.

CPSC 223b *Data Structures and Programming Techniques.*

Topics include programming in C; data structures (arrays, stacks, queues, lists, trees, heaps, graphs); sorting and searching; storage allocation and management; data abstraction; programming style; testing and debugging; writing efficient programs. *After CPSC 201a or b or equivalent.*

MATH 244a / AMTH 244a *Discrete Mathematics.*

Basic concepts and results in discrete mathematics: graphs, trees, connectivity, Ramsey theorem, enumeration, binomial coefficients, Stirling numbers. Properties of finite set systems. *After MATH 115a or b or equivalent.*