

# COMPUTER SCIENCE

---

## **CSC 109/ SDS 109 Communicating with Data (4 Credits)**

Offered as SDS 109 and CSC 109. The world is growing increasingly reliant on collecting and analyzing information to help people make decisions. Because of this, the ability to communicate effectively about data is an important component of future job prospects across nearly all disciplines. In this course, students learn the foundations of information visualization and sharpen their skills in communicating using data. Throughout the semester, we explore concepts in decision-making, human perception, color theory and storytelling as they apply to data-driven communication. Whether you're an aspiring data scientist or you just want to learn new ways of presenting information, this course helps you build a strong foundation in how to talk to people about data. {M}  
**Fall, Spring, Alternate Years**

## **CSC 110 Introduction to Computer Science (4 Credits)**

A gentle introduction to designing programs (recipes) for systematically solving problems. Students learn to build programs including designing, coding, debugging, testing and documenting them. An introduction to block-structured procedural control flow including branching, iteration and functions, using primitive and simple data types (lists). Students learn the high-level internal operation of computer systems (inputs, outputs, processing and storage) and their applications. Students are exposed to the social and historical aspects of computing. This course is recommended for those who have no prior experience in computer science at the high school, AP or college level. Not open to students who have taken CSC 111. May not be taken concurrently with CSC 120. S/U only. Enrollment limited to 30. {M}  
**Fall, Spring**

## **CSC 120 Object Oriented Programming (4 Credits)**

This course emphasizes computational problem-solving using a typed object-oriented programming (OOP). Students learn core computer science principles including: control flow, functions, classes, objects, methods, encapsulation and information-hiding, specification, recursion, debugging, unit testing, version control, using libraries and writing code in multiple files. Students also learn and apply the model-view-controller (MVC) architecture, the basics of graphics and GUIs, working with external files and foundations of algorithm design. Abstract data types and simple data structures are used to illustrate concepts of OOP and solve computational problems through regular programming assignments (in Java and Python). This course assumes prior programming experience including a basic understanding of branching (if-statements), iteration (loops), functions and simple data types (integers, strings, lists/arrays). Prerequisites: CSC 110 or equivalent. Cannot be taken concurrently with CSC 110. Enrollment limited to 30. {M}  
**Fall, Spring**

## **CSC 205/ MTH 205 Modeling in the Sciences (4 Credits)**

Offered as CSC 205 and MTH 205. This course integrates the use of mathematics and computers for modeling various phenomena drawn from the natural and social sciences. Scientific case studies span a wide range of systems at all scales, with special emphasis on the life sciences. Mathematical tools include data analysis, discrete and continuous dynamical systems, and discrete geometry. This is a project-based course and provides elementary training in programming using Mathematica. Designations: Theory, Programming. Prerequisites: MTH 112. CSC 110 recommended. Enrollment limited to 20. {M}  
**Fall, Spring, Annually**

## **CSC 210 Data Structures (4 Credits)**

Explores elementary data structures (arrays, linked lists, stacks, queues, maps, trees, graphs) and algorithms (searching, sorting, tree and graph traversal) in a variety of contexts. Using a typed object oriented programming language (e.g. Java), students develop their own implementations as well as more complex applications based upon existing, standard data structures libraries. Not open to students who have taken CSC 212. Prerequisite: CSC 120. Enrollment limited to 30. {M}  
**Fall, Spring**

## **CSC 220 Advanced Programming Techniques (4 Credits)**

Reinforces programming skills learned in previous programming courses through working on a number of projects. Offers practice for developing modular, reusable, maintainable code. Students will gain more experience with design and development. Designation: Programming. Prerequisite: CSC 210 or equivalent. {M}  
**Fall, Spring, Variable**

## **CSC 223 Introduction to Software Engineering (4 Credits)**

Introduction to software engineering theory and methodologies, with an emphasis on rapid prototyping and development. This course is a survey of topics: requirements elicitation and specification, prototyping and infrastructure, basic project management, architecture and design patterns, and verification and testing. Students work in teams on a significant design and development project. Prerequisite: CSC 210 or equivalent. Designation: Systems. Enrollment limited to 32. {M}  
**Fall, Spring**

## **CSC 230 Introduction to Database Systems (4 Credits)**

Databases form the foundation of modern commerce, social media, and government. This course will investigate the design and use of database systems from the traditional to the present day. Prerequisites: CSC 210 or equivalent. Designation: Systems. Enrollment limited to 30. {M}  
**Fall, Spring, Variable**

## **CSC 231 Microprocessors and Assembly Language (4 Credits)**

An introduction to the architecture of the Intel Pentium class processor and its assembly language in the Linux environment. Students write programs in assembly and explore the architectural features of the Pentium, including its use of the memory, the data formats used to represent information, the implementation of high-level language constructs, integer and floating-point arithmetic, and how the processor deals with I/O devices and interrupts. Prerequisite: CSC 210 or equivalent. Enrollment limited to 30. {M}  
**Fall, Spring**

## **CSC 235/ SDS 235 Visual Analytics (4 Credits)**

Offered as CSC 235 and SDS 235. Visual analytics techniques can help people to derive insight from massive, dynamic, ambiguous and often conflicting data. During this course, students learn the foundations of the emerging, multidisciplinary field of visual analytics and apply these techniques toward a focused research problem in a domain of personal interest. Students may elect to take this course as a programming intensive course, prerequisite: CSC 212. In this track, students learn to use R, Python and HTML5/JavaScript to develop custom visual analytic tools. Students preferring a non-programming intensive track may elect to use existing visual analytic software, such as Tableau or Plotly. Designations: Theory, Programming. Prerequisite: CSC 120 or equivalent. {M}  
**Fall, Spring, Variable**

**CSC 240 Computer Graphics (4 Credits)**

Covers two-dimensional drawings and transformations, three-dimensional graphics, lighting and colors, game design, perspective, curves and surfaces, ray tracing. Employs Postscript, C++, GameMaker, POV-ray, and radiosity. The course accommodates both CS majors, for whom it is programming intensive, and other students with less technical expertise, by having two tracks of assignments. Prerequisite: CSC 120 or equivalent. MTH 111 or equivalent is an additional required prerequisite for majors. Designation: Theory, Programming. {M}

**Fall, Spring, Annually**

**CSC 249 Computer Networks (4 Credits)**

This course introduces fundamental concepts in the design and implementation of computer communication networks, their protocols and applications. Topics covered include layered network architecture, physical layer and data link protocols; and transport protocols; routing protocols and applications. Most case studies are drawn from the Internet TCP/IP protocol suite. Designation: Systems. Prerequisites: CSC 120 or equivalent. {M}

**Fall**

**CSC 250 Theory of Computation (4 Credits)**

Automata and finite state machines, regular sets and regular languages, push-down automata and context-free languages, linear-bounded automata, computability and Turing machines, nondeterminism and undecidability. Prerequisites: CSC 110 and MTH 153. Enrollment limited to 30. {M}

**Fall, Spring**

**CSC 251 Network Security (4 Credits)**

This course covers a wide range of topics in network security, with a focus on both core principles and practical information. Students learn core network protocols, cryptography as information protection technologies, and the attacks and defenses most closely related to the network rather than the endpoints (e.g., laptops, desktops) on a network. Subtopics include: authentication protocols, firewalls, intrusion detection, routing and DNS security, scanning, eavesdropping, DoS attacks, PKI, password, privacy, anonymity, and recent advancements in the field. The course consists of a mixed format of lectures, in-class labs and hands-on sessions. Designation: Systems. Prerequisites: CSC 210 and CSC 249. Enrollment limited to 30.

**Spring, Variable**

**CSC 252 Algorithms (4 Credits)**

Covers algorithm design techniques ("divide-and-conquer," dynamic programming, "greedy" algorithms, etc.), analysis techniques (including big-O notation, recurrence relations), useful data structures (including heaps, search trees, adjacency lists), efficient algorithms for a variety of problems and NP-completeness. Designation: Theory. Prerequisites: CSC 210, MTH 111 and MTH 153. Enrollment limited to 30. {M}

**Fall, Spring, Alternate Years**

**CSC 262 Operating Systems (4 Credits)**

An introduction to the functions of an operating system and their underlying implementation. Topics include file systems, CPU and memory management, concurrent communicating processes, deadlock, and access and protection issues. Programming projects implement and explore algorithms related to several of these topics. Designations: Programming, Systems. Prerequisite: CSC 231. Enrollment limited to 30. {M}

**Fall, Spring, Annually**

**CSC 266 Introduction to Compiler Design (4 Credits)**

In this course, the students will learn the formal definition of programming language syntax and semantics. They will be introduced to the functions of compilers and their design and implementation details. The course will reinforce the students' knowledge of context free grammars and automata and use this knowledge in designing lexical analyzers and translators for high level programming languages. Topics covered include lexical analysis, type checking, context analysis and code generation. Designations: Theory, Programming. Prerequisites: CSC 231 and CSC 250.

**Fall, Spring, Variable**

**CSC 268 Image Processing Fundamentals (4 Credits)**

Images fill the media, and most are processed by computer at some point or another. This course will examine a variety of algorithmic image processing techniques, exploring implementation and applications, as well as some of the social impact and ethical issues surrounding their use. Prerequisites: CSC 212 and MTH 111. Enrollment limited to 30. {M}

**Fall, Spring, Variable**

**CSC 270 Digital Circuits and Computer Systems (5 Credits)**

This class introduces the operation of logic and sequential circuits. Students explore basic logic gates (AND, OR, NAND, NOR), counters, flip-flops, decoders, microprocessor systems. Students have the opportunity to design and implement digital circuits during a weekly lab. Designation: Systems. Prerequisite: CSC 231. Enrollment limited to 12.

**Fall, Spring, Variable**

**CSC 274 Discrete and Computational Geometry (4 Credits)**

Topics include the core of the field: polygons, convex hulls, triangulations and Voronoi diagrams. Beyond this core, curves and surfaces, and computational topology are covered. Throughout, a dual emphasis is maintained on mathematical proofs and efficient algorithms. Students have a choice of concentrating their course work in mathematics or toward computer science. Designations: Theory, Programming. Prerequisite for MTH major credit: MTH 153. MTH 111 recommended. Prerequisite for CSC major credit: CSC 120. {M}

**Fall, Spring, Variable**

**CSC 290 Introduction to Artificial Intelligence (4 Credits)**

An introduction to artificial intelligence including an introduction to artificial intelligence programming. Discussions include: game playing and search strategies, machine learning, natural language understanding, neural networks, genetic algorithms, evolutionary programming and philosophical issues. Designations: Theory, Programming. Prerequisite: CSC 120 or equivalent. Prerequisites for CSC Majors: CSC 210 and MTH 111 or equivalent.

**Fall, Spring, Variable**

**CSC 294 Computational Machine Learning (4 Credits)**

An introduction to machine learning from a programming perspective. Students will develop an understanding of the basic machine learning concepts (including underfitting/overfitting, measures of model complexity, training/test set splitting and cross validation), but with an explicit focus on machine learning systems design (including evaluating algorithmic complexity and development of programming architecture) and on machine learning at scale. Principles of supervised and unsupervised learning will be demonstrated via an array of machine learning methods including decision trees, k-nearest neighbors, ensemble methods and neural-networks/deep-learning as well as dimension reduction, clustering and recommender systems. Students will implement classic machine learning techniques, including gradient descent. Designations: Theory, Programming. Prerequisites: CSC 210, CSC 250 & (MTH 112 or MTH 211), and knowledge of Python. Enrollment limited to 40. {M}

**Fall, Spring, Annually**

**CSC 325 Seminar: Responsible Computing (4 Credits)**

When is disruption good? Who is responsible for ensuring that an innovation has a positive impact? Are these impacts shared equitably? How can bias be eliminated from algorithms, if they exist? What assurances can anyone make about the technology they develop? What are the limitations of professional ethics? This seminar examines the ethical implication (i.e., ethics, justice, political philosophy) of computing and automation. Participants will explore how to design technology responsibly while contributing to progress and growth. Topics include: intellectual property; privacy, security and freedom of information; automation; globalization; access to technology; artificial intelligence; mass society; and emerging issues. Designation: Systems. Prerequisite: CSC 210. Enrollment limited to 12. Juniors and seniors only. Instructor permission required. {S}

**Fall, Spring, Variable**

**CSC 327 Seminar: Internet Censorship (4 Credits)**

Internet censorship deals with practices of information control. This seminar covers topics related to 1) the existing blocking mechanisms at different layers of the Internet protocol stacks used by network censors and 2) the broad issue of detecting the type of network interference as indicative of Internet censorship. This course will include topics such as traffic differentiation, surveillance, and blocking of content. Enrollment limited to 12. Juniors and seniors only. Instructor permission required.

**Fall, Spring, Variable**

**CSC 328/ EGR 328 Seminar: Digital Circuits and Sensors (4 Credits)**

Offered as CSC 328 and EGR 328. Previously EGR 390dc. Digital circuits are everywhere, from basic thermostat controls and stop light sequencers to smart phones, computers and even Mars Rovers! This course covers the basic building blocks for all electronics. Students investigate basic logic circuits, combinatorial logic and sequential logic with an introduction to the basic digital circuits such as encoders and multiplexers. The second part of the semester focuses on microprocessors, using the Arduino. Students will build a variety of circuits with input (from a computer, or from the environment via sensors) and programmed output (LEDs, sound, data sent to a computer), in order to learn how information from our analog world can be converted into digital data. There will be a lab about every other week, and a final project for students to explore an area of their choosing in more depth. Prerequisites: CSC 110 or CSC 120, and either EGR 220 or CSC 231. Enrollment limited to 12. Junior and senior engineering majors only. {N}

**Fall, Spring, Variable**

**CSC 330 Seminar: Database Systems (4 Credits)**

This course covers principles of database systems, including such topics as data independence, storage structures, relational data models, security, and integrity. It will also touch on some non-relational database systems, and alternative consistency mechanisms. As a seminar course, it will mix theory, programming, and research. Designation: Systems. Prerequisites: CSC 210 and MTH 153. Enrollment limited to 12. Juniors and seniors only. Instructor permission required. {M}

**Fall, Spring, Variable**

**CSC 334bd Seminar: Topics in Computational Biology-Biomedical Big Data (4 Credits)**

This course explores the intersection of computer science and biomedical research. In the genomic era, biological and clinical research generates vast amounts of omics data, much of which is publicly available. Students examine the scientific literature to learn about ways that researchers are harnessing this data to make new discoveries in biomedical domains. This course also discusses the challenges that biomedical big data presents in terms of storage, access and analysis. Finally, students engage in hands-on, project-based learning where they implement the approaches discussed to mine biological repositories, develop their own algorithms and test their own hypotheses. Prerequisite: CSC 294. A working knowledge of biology strongly recommended. Enrollment limited to 12. Senior computer science majors only. Instructor permission required. {M}{N}

**Fall, Spring, Variable**

**CSC 352 Seminar: Parallel Programming (4 Credits)**

The field of high-performance computing (HPC) leverages the largest and most powerful computers on the planet to enable cutting edge scientific breakthroughs that help us understand fundamental research questions. These machines and programs push the limits of speed and scalability and require a practical understanding of the entire computing stack as well as familiarity with novel and emerging hardware platforms. In this course, students will learn and apply both the theoretical and practical aspects of the field. This will include exposure to both foundational texts and ideas along with cutting-edge research, and will culminate in a student-directed group project that leverages both the technologies and techniques learned in this course. By the end of the semester, students will be able to design, execute, and evaluate potential parallelization strategies in a target domain. Prerequisites: CSC 210 and CSC 252. Enrollment limited to 12. Juniors and seniors only. Instructor permission required.

**Fall, Spring, Variable**

**CSC 353pm Seminar: Topics in Robotics-Robotics Perception and Mapping (4 Credits)**

Robotic perception is a crucial aspect of robotics. It lets artificial agents, like robots or software automatons, gather information about their environment to make informed decisions. Perception includes sensing of light information (vision), sound information (audition), tactile or limb-position information (touch / proprioception) and several other aspects that can be integrated to give the agent a sufficient idea of its context. In this seminar, we will discuss the basics concepts, the history and future directions of robotic perception. Enrollment limited to 12. Juniors and seniors only. Instructor permission required.

**Fall, Spring, Variable**

**CSC 354 Seminar: Music Information Retrieval (4 Credits)**

This course is envisioned to serve as an introduction to the field of Music Information Retrieval (MIR), covering both theoretical and practical elements of the field. This seminar aims to prepare students for research in MIR either in academia or industry. Topics will range from feature extraction and structure tasks to debates about the latest music-based apps and questions about music licensing. The course will embody the liberal arts experience by including technical programming assignments, position papers and discussions about current research papers. Designations: Theory, Programming. Prerequisites: CSC 111 and one additional programming course, SDS 291, SDS 293 or equivalent, and one writing intensive course. Enrollment limited to 12. Juniors and seniors only. Instructor permission required.

**Fall, Spring, Variable**

**CSC 356ni Seminar: Topics in Human/Computer Interaction-Natural User Interfaces (4 Credits)**

In the study of human-computer interaction, a natural user interface (NUI) is an interaction paradigm in which the interface is largely invisible to the end user. Contrasted with WIMP ("windows, icons, menus, pointer") interfaces found in traditional computing systems, NUIs rely on seamless, intuitive interaction rather than artificial control devices. While companies such as Apple and Google have rapidly popularized the NUI model, this interaction paradigm brings with it a unique set of design challenges, constraints, and ethical considerations. In this course, we will explore ongoing research in this evolving area and put these techniques into practice in various application areas. Prerequisites: CSC 212, CSC 256 strongly recommended. {M}

**Fall, Spring, Variable**

**CSC 356pa Seminar: Topics in Human/Computer Interaction-Platform Activism (4 Credits)**

Networked platforms like social media sites, gig sharing apps, and game consoles have become important sites of study for human-computer interaction. Contemporary research on the subject includes both platform studies, which offer a critical perspective on the power that large companies have to shape the creative labor and communication patterns of their users, and technology design activism, which seeks to amplify grassroots movements for positive social change on those platforms. In this course we will explore these emerging areas of scholarship and put our learnings into practice by using participatory design to create new technological interventions on various media platforms. {M}

**Fall, Spring, Variable**

**CSC 370 Seminar: Computer Vision and Image Processing (4 Credits)**

Explores the challenge of computer vision through readings of original papers and implementation of classic algorithms. This seminar considers techniques for extracting useful information from digital images, including both the motivation and the mathematical underpinnings. Topics range from low-level techniques for image enhancement and feature detection to higher-level issues such as stereo vision, image retrieval and segmentation of tracking of objects. Designations: Theory, Programming. Prerequisites: CSC 210 and MTH 153. Enrollment limited to 12. Juniors and seniors only. Instructor permission required. {N}

**Fall, Spring, Variable**

**CSC 372 Seminar: Applied Algorithms (4 Credits)**

Covers advanced data structures and algorithms with an emphasis on object-oriented implementation, comparative efficiency analysis and applications. A variety of algorithmic approaches will be discussed (divide-and-conquer, dynamic programming, greedy algorithms, graph traversal). We'll go beyond classical searching and sorting to graphs and networks and other applied problems, as well as a selection of NP-hard ones. Designations: Theory, Programming. Prerequisites: CSC 110, CSC 210, MTH 153 and MTH 111 or equivalent. Enrollment limited to 12. Juniors and seniors only. Instructor permission required. {M}

**Fall, Spring, Variable**

**CSC 400 Special Studies (1-4 Credits)**

For majors, by arrangement with a computer science faculty member.

**Fall, Spring**

**CSC 430D Honors Project (4-8 Credits)**

**Fall, Spring, Annually**