

Computer Science

Program Description

The Department of Computer Science develops graduates who can process information in digital computers, design computer hardware and software, and work successfully with several different computing applications. The Department offers the degree of Bachelor of Science in Computer Science. Within this degree, a student may choose (not required) one of six available emphasis areas in Business, Computer Engineering, Data Science, Research Honors, Robotics and Intelligent Systems, or Space.

BS in Computer Science

Computing is ubiquitous, impacting almost every aspect of modern life, and playing an important role in many technological advances. Computing jobs are among the highest paid, and computing professionals generally report high job satisfaction. Graduates from our program have found employment with many different types of companies including technology, engineering, and financial companies.

The CS degree at Mines is designed to be accessible to students with or without prior programming experience. The Introduction to Computer Science course introduces students to the building blocks of CS and provides a brief introduction to procedural programming in Python. The second computing course, Programming Concepts, emphasizes development of programming skills in an object-oriented language. The third introductory course, Data Structures, provides an understanding of the classic data representation schemes, algorithms, and algorithm analysis that form the foundation for all advanced work in computing.

Required CS courses provide the fundamental skills and knowledge that are critical to success in computing. These courses reflect a mixture of theory and practice, including discrete structures, design and analysis of algorithms, principles of programming languages, computer architecture, operating systems, software engineering, and database management. The capstone field session course provides students an opportunity to work in teams to create software products for real clients.

Elective courses in CS allow students to explore a variety of important computing topics, such as graphics and visualization, artificial intelligence, mobile applications, and web programming. Elective courses often relate to recent trends in computing, covering topics such as security, high performance computing, and machine learning.

Computing is a broad field with applicability to most science and engineering domains. The CS minor is designed for students in other disciplines to receive a solid grounding in the basics, which should enable them to apply their computing skills to solve problems in other domains.

PROGRAM EDUCATIONAL OBJECTIVES

In addition to contributing toward achieving the educational objectives described in the Mines' Graduate Profile, the Computer Science Program at Mines has established the following program educational objectives:

Students will demonstrate technical expertise within computer science by:

- Designing and implementing solutions to practical problems in science and engineering,

- Using appropriate technology as a tool to solve problems in computer science, and
- Creating efficient algorithms and well-structured computer programs.

Students will demonstrate a breadth and depth of knowledge within computer science by:

- Extending course material to solve original problems,
- Applying knowledge of computer science to the solution of problems, and
- Identifying, formulating and solving computer science problems.

Students will demonstrate an understanding and appreciation for the relationship of computer science to other fields by:

- Applying computer science to solve problems in other fields,
- Working in cooperative multidisciplinary teams, and
- Choosing appropriate technology to solve problems in other disciplines.

Students will demonstrate an ability to communicate computer science effectively by:

- Giving oral presentations,
- Completing written explanations,
- Interacting effectively in cooperative teams,
- Creating well-documented programs, and
- Understanding and interpreting written material in computer science.

Student Learning Outcomes

1. An ability to identify, formulate, and solve complex engineering problems by applying principles of engineering, science, and mathematics.
2. An ability to apply engineering design to produce solutions that meet specified needs with consideration of public health, safety, and welfare, as well as global, cultural, social, environmental, and economic factors.
3. An ability to communicate effectively with a range of audiences.
4. An ability to recognize ethical and professional responsibilities in engineering situations and make informed judgments, which must consider the impact of engineering solutions in global, economic, environmental, and societal contexts.
5. An ability to function effectively on a team whose members together provide leadership, create a collaborative and inclusive environment, establish goals, plan tasks, and meet objectives.
6. An ability to develop and conduct appropriate experimentation, analyze and interpret data, and use engineering judgment to draw conclusions.
7. An ability to acquire and apply new knowledge as needed, using appropriate learning strategies.

Bachelor of Science in Computer Science Degree Requirements:

Freshman

Fall		lec	lab	sem.hrs
MATH111	CALCULUS FOR SCIENTISTS AND ENGINEERS I			4.0
HASS100	NATURE AND HUMAN VALUES			3.0

CHGN121	PRINCIPLES OF CHEMISTRY I	4.0
CSM101	FRESHMAN SUCCESS SEMINAR	1.0
CSCI128	COMPUTER SCIENCE FOR STEM	3.0

15.0

Spring			lec	lab	sem.hrs
MATH112	CALCULUS FOR SCIENTISTS AND ENGINEERS II	4.0			
PHGN100	PHYSICS I - MECHANICS	4.0			
S&W	SUCCESS AND WELLNESS	1.0			
EDNS151	CORNERSTONE - DESIGN I	3.0			
CSCI200	FOUNDATIONAL PROGRAMMING CONCEPTS & DESIGN	3.0			

15.0**Sophomore**

Fall			lec	lab	sem.hrs
MATH213	CALCULUS FOR SCIENTISTS AND ENGINEERS III	4.0			
CSM202	INTRODUCTION TO STUDENT WELL-BEING AT MINES	1.0			
PHGN200	PHYSICS II- ELECTROMAGNETISM AND OPTICS	4.0			
CSCI358	DISCRETE MATHEMATICS	3.0			
CSCI210	SYSTEMS PROGRAMMING	3.0			

15.0

Spring			lec	lab	sem.hrs
MATH225	DIFFERENTIAL EQUATIONS	3.0			
HASS200	GLOBAL STUDIES	3.0			
FREE	Free Elective	3.0			
CSCI220	DATA STRUCTURES AND ALGORITHMS	3.0			
Focus Area Course 1		3.0			

15.0**Junior**

Fall			lec	lab	sem.hrs
MATH332	LINEAR ALGEBRA	3.0			
MATH334	INTRODUCTION TO PROBABILITY	3.0			
CSCI306	SOFTWARE ENGINEERING	3.0			
Focus Area Course 2		3.0			
Focus Area Course 3		3.0			

15.0

Spring		lec	lab	sem.hrs
EBGN321	ENGINEERING ECONOMICS ^{*For the 2023 Catalog} EBGN321 replaced EBGN201 as a Core requirement. EBGN321 was added to the core, but has a prerequisite of 60 credit hours. Students whose programs that required EBGN201 the sophomore year may need to wait to take EBGN321 until their junior year. For complete details, please visit: https://www.mines.edu/registrar/core-curriculum/			3.0
ELECTIVE	CULTURE AND SOCIETY (CAS) Mid-Level Restricted Elective			3.0
FREE	Free Elective			3.0
CSCI341	COMPUTER ORGANIZATION			3.0
CSCI406	ALGORITHMS			3.0
Focus Area Course 4				3.0

18.0

Summer		lec	lab	sem.hrs
CSCI370	ADVANCED SOFTWARE ENGINEERING			5.0

5.0**Senior**

Fall		lec	lab	sem.hrs
ELECTIVE	CULTURE AND SOCIETY (CAS) Mid-Level Restricted Elective			3.0
CSCI400	PRINCIPLES OF PROGRAMMING LANGUAGES			3.0
Focus Area Course 5				3.0
Focus Area Course 6				3.0
Focus Area Course 7				3.0

15.0

Spring		lec	lab	sem.hrs
ELECTIVE	CULTURE AND SOCIETY (CAS) 400 Level Restricted Elective			3.0
CSCI442	OPERATING SYSTEMS			3.0
Focus Area Course 8				3.0
Focus Area Course 9				3.0
Focus Area Course 10				3.0

15.0**Total Semester Hrs: 128.0****Focus Areas**

The Department of Computer Science offers seven focus areas:

1. Computer Science
2. CS + Business
3. CS + Computer Engineering
4. CS + Data Science
5. CS + Research Honors

6. CS + Robotics & Intelligent Systems
7. CS + Space

Computer Science Electives can be chosen from any CSCI400-level course, any CSCI500-level course (with advisor approval), MATH307, or EENG383. EDNS491 & EDNS492, when both courses are taken together, can both be counted as Computer Science Electives. In a given focus area, a required course for that focus area can not also be counted as a CSCI technical elective in that focus area.

Computer Science

FREE	Free Elective	3.0
CSCI403	DATA BASE MANAGEMENT	3.0
FREE	Free Elective	3.0
CSCI ELECT	Computer Science Elective	3.0
CSCI ELECT	Computer Science Elective	3.0
CSCI ELECT	Computer Science Elective	3.0
FREE	Free Elective	3.0
CSCI ELECT	Computer Science Elective	3.0
CSCI ELECT	Computer Science Elective	3.0
FREE	Free Elective	3.0
Total Semester Hrs		30.0

CS + Business

CSCI303	INTRODUCTION TO DATA SCIENCE	3.0
CSCI403	DATA BASE MANAGEMENT	3.0
EBGN***	BS Elective #&	3.0
EBGN***	BS Elective #&	3.0
EBGN***	BS Elective #&	3.0
CSCI475	INFORMATION SECURITY AND PRIVACY	3.0
CSCI ELECT	Computer Science Elective	3.0
EBGN***	BS Elective #&	3.0
CSCI448	MOBILE APPLICATION DEVELOPMENT	3.0
or CSCI445	WEB PROGRAMMING	
or CSCI446	WEB APPLICATIONS	
FREE	Free Elective	3.0
Total Semester Hrs		30.0

Four unique Business Electives must be chosen from:

EBGN230, EBGN320, EBGN321, EBGN345, EBGN346, EBGN360, EBGN420, EBGN460, EBGN485

& If Business Elective is used to satisfy CAS Restricted Elective requirement, then an additional Business Elective or eligible Computer Science Elective must be substituted.

CS + Computer Engineering

EENG281	INTRODUCTION TO ELECTRICAL CIRCUITS, ELECTRONICS AND POWER #&	3-4
or EENG282	ELECTRICAL CIRCUITS	
or PHGN215	ANALOG ELECTRONICS	
EENG284	DIGITAL LOGIC #	3-4
or PHGN317	SEMICONDUCTOR CIRCUITS- DIGITAL	
EENG383	EMBEDDED SYSTEMS	4.0
CSCI471	COMPUTER NETWORKS I	3.0
CSCI475	INFORMATION SECURITY AND PRIVACY	3.0

CSCI ELECT	Computer Science Elective	3.0
CSCI403	DATA BASE MANAGEMENT	3.0
or CSCI440	PARALLEL COMPUTING FOR SCIENTISTS AND ENGINEERS	
or CSCI474	INTRODUCTION TO CRYPTOGRAPHY	
CSCI/EENG ELECT	Computer Science or Electrical Engineering Elective ^	3.0
FREE	Free Elective	3.0
FREE	Free Elective	1.0
Total Semester Hrs		29-31

PHGN215 & PHGN317 provide similar content to EENG281 & EENG284 and both can be substituted together with preapproval.

& EENG282 may be substituted for EENG281

^ EE elective may be chosen from any EENG 400-level course

CS + Data Science

CSCI303	INTRODUCTION TO DATA SCIENCE	3.0
CSCI403	DATA BASE MANAGEMENT	3.0
MATH201	INTRODUCTION TO STATISTICS	3.0
MATH335	INTRODUCTION TO MATHEMATICAL STATISTICS	3.0
MATH324	STATISTICAL MODELING	3.0
CSCI470	INTRODUCTION TO MACHINE LEARNING	3.0
MATH432	SPATIAL STATISTICS	3.0
or MATH436	ADVANCED STATISTICAL MODELING	
or MATH437	MULTIVARIATE ANALYSIS	
or MATH438	STOCHASTIC MODELS	
or MATH439	SURVIVAL ANALYSIS	
CSCI404	ARTIFICIAL INTELLIGENCE	3.0
or CSCI423	COMPUTER SIMULATION	
or CSCI440	PARALLEL COMPUTING FOR SCIENTISTS AND ENGINEERS	
or CSCI474	INTRODUCTION TO CRYPTOGRAPHY	
or CSCI475	INFORMATION SECURITY AND PRIVACY	
CSCI ELECT	Computer Science Elective	3.0
FREE	Free Elective	3.0
Total Semester Hrs		30.0

CS + Research Honors

FREE	Free Elective	3.0
CSCI403	DATA BASE MANAGEMENT	3.0
HASS360	RESEARCH, VALUES, AND COMMUNICATION	3.0
CSCI ELECT	Computer Science Elective	3.0
CSCI480	COMPUTER SCIENCE HONORS THESIS	3.0
CSCI5**	Computer Science Honors Elective #	3.0
CSCI ELECT	Computer Science Elective	3.0
CSCI480	COMPUTER SCIENCE HONORS THESIS	3.0
CSCI5**	Computer Science Honors Elective #	3.0
CSCI ELECT	Computer Science Elective	3.0
Total Semester Hrs		30.0

Computer Science Honors Electives may be any CSCI500-level course approved by advisor.

CS + Robotics & Intelligent Systems

EENG281	INTRODUCTION TO ELECTRICAL CIRCUITS, ELECTRONICS AND POWER #&	3-4
or EENG282	ELECTRICAL CIRCUITS	
or PHGN215	ANALOG ELECTRONICS	
FREE	Free Elective	3.0
EENG307	INTRODUCTION TO FEEDBACK CONTROL SYSTEMS	3.0
MEGN441	INTRODUCTION TO ROBOTICS	3.0
CSCI404	ARTIFICIAL INTELLIGENCE	3.0
CSCI437	INTRODUCTION TO COMPUTER VISION	3.0
or CSCI473	ROBOT PROGRAMMING AND PERCEPTION	
CSCI470	INTRODUCTION TO MACHINE LEARNING	3.0
or CSCI534	ROBOT PLANNING AND MANIPULATION	
CSCI432	ROBOT ETHICS	3.0
or CSCI436	HUMAN-ROBOT INTERACTION	
CSCI473	ROBOT PROGRAMMING AND PERCEPTION	3.0
or CSCI432	ROBOT ETHICS	
or CSCI436	HUMAN-ROBOT INTERACTION	
or CSCI437	INTRODUCTION TO COMPUTER VISION	
or CSCI470	INTRODUCTION TO MACHINE LEARNING	
or CSCI534	ROBOT PLANNING AND MANIPULATION	
CSCI ELECT	Computer Science Elective	3.0
Total Semester Hrs		30-31

PHGN215 provides similar content to EENG281 and can be substituted with preapproval

& EENG282 may be substituted for EENG281

CS + Space

EENG281	INTRODUCTION TO ELECTRICAL CIRCUITS, ELECTRONICS AND POWER #&	3.0
or EENG282	ELECTRICAL CIRCUITS	
or PHGN215	ANALOG ELECTRONICS	
CSCI303	INTRODUCTION TO DATA SCIENCE	3.0
or CSCI404	ARTIFICIAL INTELLIGENCE	
or CSCI475	INFORMATION SECURITY AND PRIVACY	
FREE	Free Elective	3.0
EENG284	DIGITAL LOGIC #	4.0
or PHGN317	SEMICONDUCTOR CIRCUITS- DIGITAL	
MEGN455	AEROSPACE SYSTEMS ENGINEERING	3.0
MEGN456	SPACE OPERATIONS AND MISSION DESIGN	3.0
MEGN458	INTRO TO SPACE EXPLORATION AND RESOURCES	3.0
CSCI423	COMPUTER SIMULATION	3.0
or CSCI440	PARALLEL COMPUTING FOR SCIENTISTS AND ENGINEERS	
or CSCI471	COMPUTER NETWORKS I	

EENG383	EMBEDDED SYSTEMS	4.0
CSCI***	CS Elective	3.0

PHGN215 & PHGN317 provide similar content to EENG281 & EENG284 and both can be substituted together with preapproval.

& EENG282 may be substituted for EENG281

Major GPA

During the 2016-2017 Academic Year, the Undergraduate Council considered the policy concerning required major GPAs and which courses are included in each degree's GPA. While the GPA policy has not been officially updated, in order to provide transparency, council members agreed that publishing the courses included in each degree's GPA is beneficial to students.

The following list details the courses that are included in the GPA for this degree:

- CSCI200 through CSCI799 inclusive, excluding CSCI*99

Computer Science

For a **Minor in Computer Science**, the student needs to take:

CSCI200	FOUNDATIONAL PROGRAMMING CONCEPTS & DESIGN	3.0
CSCI210	SYSTEMS PROGRAMMING	3.0
CSCI220	DATA STRUCTURES AND ALGORITHMS	3.0
CSCI358	DISCRETE MATHEMATICS	3.0
or MATH334	INTRODUCTION TO PROBABILITY	
CSCI306	SOFTWARE ENGINEERING *	3.0
	or CSCI Elective	
CSCI ELECT	Computer Science Elective *	3.0

* CSCI Electives can be chosen from any 400-level or 500-level CSCI course, MATH307, and EENG383.

Minor in Computer Engineering

To earn the Minor in Computer Engineering, a student must take at least 18 credits from the following list, at least 9 of which must be 300-level or above:

CSCI200	FOUNDATIONAL PROGRAMMING CONCEPTS & DESIGN	3.0
CSCI210	SYSTEMS PROGRAMMING	3.0
CSCI220	DATA STRUCTURES AND ALGORITHMS	3.0
CSCI341	COMPUTER ORGANIZATION	3.0
CSCI442	OPERATING SYSTEMS	3.0
EENG281	INTRODUCTION TO ELECTRICAL CIRCUITS, ELECTRONICS AND POWER ^	3.0
or EENG282	ELECTRICAL CIRCUITS	
EENG284	DIGITAL LOGIC *	4.0
EENG383	EMBEDDED SYSTEMS	4.0

^ PHGN215 may be used in place of EENG281/EENG282 with pre-approval by the student's major program director.

* PHGN317 may be used in place of EENG284 with pre-approval by the student's major program director.

At most 6.0 credits of this minor can be counted toward another minor degree program.

Minor in Data Science

Complete one of the following sets of courses:

CSCI200	FOUNDATIONAL PROGRAMMING CONCEPTS & DESIGN	3.0
CSCI220	DATA STRUCTURES AND ALGORITHMS	3.0
CSCI303	INTRODUCTION TO DATA SCIENCE	3.0
MATH201	INTRODUCTION TO STATISTICS	3.0
Choose 2 of		
CSCI403	DATA BASE MANAGEMENT	6.0
or CSCI404	ARTIFICIAL INTELLIGENCE	
or CSCI470	INTRODUCTION TO MACHINE LEARNING	
or CSCI478	INTRODUCTION TO BIOINFORMATICS	
Total Semester Hrs		18.0

Or

CSCI200	FOUNDATIONAL PROGRAMMING CONCEPTS & DESIGN	3.0
CSCI220	DATA STRUCTURES AND ALGORITHMS	3.0
CSCI303	INTRODUCTION TO DATA SCIENCE	3.0
MATH201	INTRODUCTION TO STATISTICS	3.0
MATH334	INTRODUCTION TO PROBABILITY	3.0
MATH335	INTRODUCTION TO MATHEMATICAL STATISTICS	3.0
Total Semester Hrs		18.0

Minor in Robotics and Intelligent Systems

Complete the following courses:

CSCI200	FOUNDATIONAL PROGRAMMING CONCEPTS & DESIGN	3.0
CSCI220	DATA STRUCTURES AND ALGORITHMS	3.0
MATH334	INTRODUCTION TO PROBABILITY	3.0
MEGN441	INTRODUCTION TO ROBOTICS	3.0
CSCI404	ARTIFICIAL INTELLIGENCE	3.0
or CSCI437	INTRODUCTION TO COMPUTER VISION	
or CSCI470	INTRODUCTION TO MACHINE LEARNING	
CSCI473	ROBOT PROGRAMMING AND PERCEPTION	3.0
or CSCI436	HUMAN-ROBOT INTERACTION	
or CSCI534	ROBOT PLANNING AND MANIPULATION	
Total Semester Hrs		18.0

Courses

CSCI101. INTRODUCTION TO COMPUTER SCIENCE. 3.0 Semester Hrs.

Introduction to Computer Science is a 3-credit hour **breadth** CS course. We cover several topics in this course to help students understand how computers work, e.g., binary numbers, Boolean logic and gates, circuit design, machine language, computer hardware, assembly, operating systems, networking, the Internet protocols, cybersecurity, data science, machine learning, and robotics.

CSCI102. INTRODUCTION TO COMPUTER SCIENCE - LAB. 1.0 Semester Hr.

CSCI 102 is our Introduction to Computer Science LAB course. CSCI 102 is a 1-credit hour programming course in Python that is (A) extremely valuable for those who have never programmed and (B) required for some majors (e.g., MechE). While CSCI 102 is not required for some majors, students with little (or no) prior programming experience are strongly encouraged to enroll.

CSCI128. COMPUTER SCIENCE FOR STEM. 3.0 Semester Hrs.

Introduction to programming. Intended for students with no prior experience. Teaches basic programming constructs including data types, conditionals, loops, file I/O, functions, and objects in Python 3. Also covers topics vital to STEM computing, such as data science, best practices for code development, and software ethics.

CSCI195. CS@MINES BRIDGE SEMINAR COURSE. 1.0 Semester Hr.

The purpose of this course is to support Bridge students for success in CS@Mines. Through this course, students will: 1. Engage in activities that show how computing changes the world and impacts daily lives, 2. Delve into a some of the foundational computer science topics (e.g., Binary Numbers, Networking, Operating Systems, Cybersecurity, Cyber Physical Systems, Artificial Intelligence, Machine Learning, Data Science, Bioinformatics, Robotics), and 3. Explore different career paths in the computer science industry. Most importantly, this course will offer students the opportunity to build relationships with their cohort and their program advisors (e.g., Bridge Program Director, Graduate Program Manger), as well as be a source of strength for each other. Prerequisite: Current CS@Mines Bridge Student .

CSCI198. SPECIAL TOPICS. 1-6 Semester Hr.

(I, II) Pilot course or special topics course. Topics chosen from special interests of instructor(s) and student(s). Usually the course is offered only once. Prerequisite: none. Variable credit; 1 to 6 credit hours. Repeatable for credit under different titles.

CSCI199. INDEPENDENT STUDY. 1-6 Semester Hr.

(I, II) Individual research or special problem projects supervised by a faculty member, when a student and instructor agree on a subject matter, content, and credit hours. Prerequisite: "Independent Study" form must be completed and submitted to the Registrar. Variable credit; 1 to 6 credit hours. Repeatable for credit.

CSCI200. FOUNDATIONAL PROGRAMMING CONCEPTS & DESIGN. 3.0 Semester Hrs.

This course teaches students C++, how to manage memory properly & efficiently at run time, the principles of object-oriented programming, and how to create an algorithm using data structures & abstraction to solve a problem. Recursive data structures & algorithms will be constructed & analyzed when solving problems. Initial principal components of software engineering and design will be introduced and used when drafting a solution to a problem. Programs will be developed using a command line interface. Prerequisite: CSCI101 or CSCI128.

CSCI210. SYSTEMS PROGRAMMING. 3.0 Semester Hrs.

The Systems Programming course will teach students how to become proficient with using a Linux operating system from the command line and programming Linux systems. Topics will include: shell scripts; compiling and linking programs; redirecting input and output; controlling jobs from the command line; using advanced SSH functions such as port forwarding and dynamic proxying; file system hierarchy; package management; kernel compilation; network management; C programming with dynamic memory management, function pointers, c-style polymorphism, recursive functions, and data structures; learning how to use a code repository maintenance tool, such as git, from the command line effectively; security, privacy, and encryption concepts; inter-process communication and client-server architectures. Prerequisite: CSCI200.

CSCI220. DATA STRUCTURES AND ALGORITHMS. 3.0 Semester Hrs.

This course teaches students the design and construction of data structures such as hash tables, trees, heaps, and graphs, analysis of operations on data structures, and design and analysis of algorithms on data structures such as graph search and minimum spanning tree algorithms. Applications of data structures and algorithms on them are discussed in the context of computer systems. Students will further refine programming skills in C++ by producing software implementations of selected data structures and algorithms. Prerequisite: CSCI200 with a C- or better.

CSCI250. PYTHON-BASED COMPUTING: BUILDING A SENSOR SYSTEM. 3.0 Semester Hrs.

This course will teach students the skills needed for data collection, analysis, and visualization on a small embedded device (e.g., Raspberry Pi). Students will learn basic Linux, Python, and the programming skills needed to control the hardware and associated sensors. This hands-on course includes a baseline project, four introductory projects (e.g., acoustic, acceleration, magnetic field, optical), and a final Capstone project. The Capstone project will have students create their own application using the techniques learned during the first half of the semester; students will then present their Capstone project through a formal presentation, write-up, and demonstration. We suggest the student take "Introduction to Computer Science" before this course. Prerequisite: CSCI128. Co-requisite: MATH213, PHGN200.

CSCI260. FORTRAN PROGRAMMING. 2.0 Semester Hrs.

(I) Computer programming in Fortran90/95 with applications to science and engineering. Program design and structure, problem analysis, debugging, program testing. Language skills: arithmetic, input/output, branching and looping, functions, arrays, data types. Introduction to operating systems. Prerequisite: none. 2 hours lecture; 2 semester hours.

CSCI261. PROGRAMMING CONCEPTS. 3.0 Semester Hrs.

This course introduces fundamental computer programming concepts using a high-level language and a modern development environment. Programming skills include sequential, selection, and repetition control structures, functions, input and output, primitive data types, basic data structures including arrays and pointers, objects, and classes. Software engineering skills include problem solving, program design, and debugging practices. Prerequisite: CSCI101.

CSCI262. DATA STRUCTURES. 3.0 Semester Hrs.

(I, II, S) Defining and using data structures such as linked lists, stacks, queues, binary trees, binary heap, and hash tables. Introduction to algorithm analysis, with emphasis on sorting and search routines. Language skills: abstract data types, templates, and inheritance. 3 hours lecture; 3 semester hours. Prerequisite: CSCI261 with a grade of C- or higher.

CSCI274. INTRODUCTION TO THE LINUX OPERATING SYSTEM. 1.0 Semester Hr.

(I, II) Introduction to the Linux Operating System will teach students how to become proficient with using a Linux operating system from the command line. Topics will include: remote login (ssh), file system navigation, file commands, editors, compilation, execution, redirection, output, searching, processes, usage, permissions, compression, parsing, networking, and bash scripting. Prerequisite: CSCI200 or CSCI261.

CSCI290. PROGRAMMING CHALLENGES I. 1.0 Semester Hr.

This course is the first of three courses in the Programming Challenges sequence which covers problem solving patterns and paradigms found in technical interviews and programming competitions. The students will learn more advanced data structures and algorithms while focusing on algorithmic complexity to solve problems in a finite amount of time. Co-requisite: CSCI220 or CSCI262.

CSCI295. INDUSTRY EXPLORATION I. 1.0 Semester Hr.

Industry Exploration I provides 1st and 2nd year students an opportunity to explore different career paths in computer science. Each week students meet (over Zoom) with a company that hires a number of computer scientists. Prior to the meeting, students research the company and determine 1-2 specific question(s) to ask during the meeting (i.e., questions specific to the company). During the meeting, students talk with employees at the company to learn more about the types of computer science jobs that exist. After the meeting, students reflect on what was learned during the meeting. At the end of the semester, students have a better understanding of different types of jobs in computer science and what they may want to do in their future careers.

CSCI298. SPECIAL TOPICS. 1-6 Semester Hr.

(I, II) Pilot course or special topics course. Topics chosen from special interests of instructor(s) and student(s). Usually the course is offered only once. Prerequisite: none. Variable credit; 1 to 6 credit hours. Repeatable for credit under different titles.

CSCI299. INDEPENDENT STUDY. 1-6 Semester Hr.

(I, II) Individual research or special problem projects supervised by a faculty member, when a student and instructor agree on a subject matter, content, and credit hours. Prerequisite: "Independent Study" form must be completed and submitted to the Registrar. Variable credit; 1 to 6 credit hours. Repeatable for credit.

CSCI303. INTRODUCTION TO DATA SCIENCE. 3.0 Semester Hrs.

This course will teach students the core skills needed for gathering, cleaning, organizing, analyzing, interpreting, and visualizing data. Students will learn basic SQL for working with databases, basic Python programming for data manipulation, and the use and application of statistical and machine learning toolkits for data analysis. The course will be primarily focused on applications, with an emphasis on working with real (non-synthetic) datasets. Prerequisite: CSCI101 or CSCI102 or CSCI128 or CSCI200 or CSCI261.

CSCI306. SOFTWARE ENGINEERING. 3.0 Semester Hrs.

(I, II) Introduction to software engineering processes and object-oriented design principles. Topics include the Agile development methodology, test-driven development, UML diagrams, use cases and several object-oriented design patterns. Course work emphasizes good programming practices via version control and code reviews. Prerequisite: CSCI210 with grade of C- or higher, CSCI220 with grade of C- or higher or CSCI262 with grade of C- or higher.

CSCI340. COOPERATIVE EDUCATION. 3.0 Semester Hrs.

(I, II, S) (WI) Supervised, full-time engineering-related employment for a continuous six-month period (or its equivalent) in which specific educational objectives are achieved. Prerequisite: Second semester sophomore status and a cumulative grade point average of at least 2.00. 0 to 3 semester hours. Cooperative Education credit does not count toward graduation except under special conditions. Repeatable.

CSCI341. COMPUTER ORGANIZATION. 3.0 Semester Hrs.

(I, II) Covers the basic concepts of computer architecture and organization. Topics include machine level instructions and operating system calls used to write programs in assembly language, computer arithmetics, performance, processor design, and pipelining techniques. This course provides insight into the way computers operate at the machine level. Prerequisite: CSCI200 or CSCI261, CSCI262.

CSCI358. DISCRETE MATHEMATICS. 3.0 Semester Hrs.

This course is an introductory course in discrete mathematics and algebraic structures. Topics include: formal logic; proofs, recursion, analysis of algorithms; sets and combinatorics; relations, functions, and matrices; Boolean algebra and computer logic; trees, graphs, finite-state machines, and regular languages. Prerequisite: MATH112 or MATH113 or MATH122.

CSCI370. ADVANCED SOFTWARE ENGINEERING. 5.0 Semester Hrs.

This capstone course has three primary goals: (1) to enable students to apply their course work knowledge to a challenging applied problem for a real client, (2) to enhance students' verbal and written communication skills, and (3) to provide an introduction to ethical decision making in computer science. Ethics and communication skills are emphasized in a classroom setting. The client work is done in small teams, either on campus or at the client site. Faculty advisors provide guidance related to the software engineering process, which is similar to Scrum. By the end of the course students must have a finished product with appropriate documentation. Prerequisite: CSCI306.

CSCI390. PROGRAMMING CHALLENGES II. 1.0 Semester Hr.

This course is the second of three courses in the Programming Challenges sequence which covers problem solving patterns and paradigms found in technical interviews and programming competitions. The students will learn more advanced set, counting, & number theory and algorithms while focusing on algorithmic complexity to solve problems in a finite amount of time. Prerequisite: CSCI290. Co-requisite: CSCI358.

CSCI395. INDUSTRY EXPLORATION II. 1.0 Semester Hr.

Industry Exploration II provides 3rd and 4th year students an opportunity to explore different career paths in computer science. Each week students visit a company that hires a number of computer scientists at the company's office. Prior to the visit, students research the company and determine 1-2 specific question(s) to ask during the meeting (i.e., questions specific to the company). During the visit, students are provided a tour of the company's office and the opportunity to talk to employees and learn more about the types of computer science jobs that exist. After the visit, students reflect on what was learned during the visit. At the end of the semester, students have a better understanding of different types of jobs in computer science and what they may want to do in their future careers.

CSCI398. SPECIAL TOPICS. 1-6 Semester Hr.

(I, II) Pilot course or special topics course. Topics chosen from special interests of instructor(s) and student(s). Usually the course is offered only once. Prerequisite: none. Variable credit; 1 to 6 credit hours. Repeatable for credit under different titles.

CSCI399. INDEPENDENT STUDY. 1-6 Semester Hr.

(I, II) Individual research or special problem projects supervised by a faculty member, when a student and instructor agree on a subject matter, content, and credit hours. Prerequisite: "Independent Study" form must be completed and submitted to the Registrar. Variable credit; 1 to 6 credit hours. Repeatable for credit.

CSCI400. PRINCIPLES OF PROGRAMMING LANGUAGES. 3.0 Semester Hrs.

This course takes a broad view of programming languages, focusing on the fundamental abstractions and principles of language design that transcend the specifics of any particular programming language. The course will emphasize functional programming, develop experience via programming projects, and cover topics such as lambda calculus, higher-order functions, induction, persistence, type systems, syntax, and evaluation. Ultimately, students will have an opportunity to improve programming skills, and develop a deeper understanding of how programming languages are designed and implemented. Prerequisite: CSCI306, CSCI358.

CSCI403. DATA BASE MANAGEMENT. 3.0 Semester Hrs.

(I, II) Design and evaluation of information storage and retrieval systems, including defining and building a database and producing the necessary queries for access to the stored information. Relational database management systems, structured query language, and data storage facilities. Applications of data structures such as lists, inverted lists and trees. System security, maintenance, recovery and definition. Interfacing host languages to database systems and object-relational mapping tools. NoSQL databases and distributed databases. Prerequisite: CSCI200 with a grade of C- or higher or CSCI262 with a grade of C- or higher.

CSCI404. ARTIFICIAL INTELLIGENCE. 3.0 Semester Hrs.

General investigation of the Artificial Intelligence field. Several methods used in artificial intelligence such as search strategies, knowledge representation, logic and probabilistic reasoning are developed and applied to practical problems. Fundamental artificial intelligence techniques are presented, including neural networks, genetic algorithms, and fuzzy sets. Selected application areas, such as robotics, natural language processing and games, are discussed. Prerequisite: CSCI220 with a grade of C- or higher or CSCI262 with a grade of C- or higher, MATH201 or MATH334.

CSCI406. ALGORITHMS. 3.0 Semester Hrs.

Equivalent with MATH406,

(I, II) Reasoning about algorithm correctness (proofs, counterexamples). Analysis of algorithms: asymptotic and practical complexity. Review of dictionary data structures (including balanced search trees). Priority queues. Advanced sorting algorithms (heapsort, radix sort). Advanced algorithmic concepts illustrated through sorting (randomized algorithms, lower bounds, divide and conquer). Dynamic programming. Backtracking. Algorithms on unweighted graphs (traversals) and weighted graphs (minimum spanning trees, shortest paths, network flows and bipartite matching); NP-completeness and its consequences. Prerequisite: CSCI220 with a grade of C- or higher or CSCI262 with a grade of C- or higher, MATH213 or MATH223 or MATH224, MATH300 or MATH358 or CSCI358.

CSCI410. ELEMENTS OF COMPUTING SYSTEMS. 3.0 Semester Hrs.

(I, II) This comprehensive course will help students consolidate their understanding of all fundamental computer science concepts. Topics include symbolic communication, Boolean logic, binary systems, logic gates, computer architecture, assembly language, assembler construction, virtual machines, object-oriented programming languages, software engineering, compilers, language design, and operating systems. Using a hardware simulator and a programming language of their choice, students construct an entire modern computer from the ground up, resulting in an intimate understanding of how each component works. Prerequisites: CSCI341 or EENG383. 3 lecture hours, 3 credit hours.

CSCI422. USER INTERFACES. 3.0 Semester Hrs.

(I) User Interface Design is a course for programmers who want to learn how to create more effective software. This objective will be achieved by studying principles and patterns of interaction design, critiquing existing software using criteria presented in the textbooks, and applying criteria to the design and implementation of one larger product. Students will also learn a variety of techniques to guide the software design process, including Cognitive Walkthrough, Talk-aloud and others. Prerequisite: CSCI262. 3 hours lecture; 3 semester hours.

CSCI423. COMPUTER SIMULATION. 3.0 Semester Hrs.

A first course in computer simulation using formal learning groups and emphasizing the rigorous development of simulation applications. Topics will include random number generation, Monte Carlo simulation, discrete event simulation, and the mathematics behind their proper implementation and analysis (random variates, arrival time modeling, infinite horizon statistics, batch means and sampling techniques). The course uses learning group assignments, quizzes, programming projects (using Linux) and exams for assessment. Prerequisite: (CSCI210 or CSCI274) AND CSCI306 AND (MATH201 or MATH334).

CSCI425. COMPILER DESIGN. 3.0 Semester Hrs.

An introductory course to the design and construction of compilers. Topics include scanning (lexical analysis), context free grammars, recursive descent (top-down) parsing, LR (bottom up) parsing, syntax directed translation, syntax trees, expression trees, parse trees, intermediate representation, register allocation and target code generation. Students will construct their own tool chain for compiling a simple language, tracking the relevant course topics as they are covered. Prerequisite: CSCI274, CSCI306, CSCI341.

CSCI432. ROBOT ETHICS. 3.0 Semester Hrs.

(II) (WI) This course explores ethical issues arising in robotics and human-robot interaction through philosophical analysis, behavioral and psychological analysis, research ethics education, and the integration of social and ethical concerns in scientific experimentation and algorithm design. Topics include case studies in lethal autonomous weapon systems, autonomous cars, and social robots, as well as higher-level concerns including economics, law, policy, and discrimination. Prerequisite: CSCI220 or CSCI262 and MATH201.

CSCI436. HUMAN-ROBOT INTERACTION. 3.0 Semester Hrs.

Human-Robot Interaction is an interdisciplinary field at the intersection of Computer Science, Robotics, Psychology, and Human Factors, that seeks to answer a broad set of questions about robots designed to interact with humans (e.g., assistive robots, educational robots, and service robots), such as: (1) How does human interaction with robots differ from interaction with other people? (2) How does the appearance and behavior of a robot change how humans perceive, trust, and interact with that robot? And (3) How can we design and program robots that are natural, trustworthy, and effective? Accordingly, In this course, students will learn (1) how to design interactive robots, (2) the algorithmic foundations of interactive robots; and (3) how to evaluate interactive robots. To achieve these learning objectives, students will read and present key papers from the HRI literature, and complete a final project in which they will design, pilot, and evaluate novel HRI experiments in small groups, with in-class time expected to be split between lecture by the instructor, presentations by students, and either collaborative active learning activities or discussions with researchers in the field. Prerequisite: CSCI200 or CSCI262 and MATH201.

CSCI437. INTRODUCTION TO COMPUTER VISION. 3.0 Semester Hrs.

Equivalent with CSCI512,EENG507,EENG512,

(I) Computer vision is the process of using computers to acquire images, transform images, and extract symbolic descriptions from images. This course provides an introduction to this field, covering topics in image formation, feature extraction, location estimation, and object recognition. Design ability and hands-on projects will be emphasized, using popular software tools. The course will be of interest both to those who want to learn more about the subject and to those who just want to use computer imaging techniques. Must be Senior level standing. 3 hours lecture; 3 semester hours. Prerequisite: (MATH201 or MATH334 or EENG311),and,MATH332,and,(CSCI200 or CSCI261).

CSCI440. PARALLEL COMPUTING FOR SCIENTISTS AND ENGINEERS. 3.0 Semester Hrs.

Equivalent with MATH440,

(II) This course is designed to introduce the field of parallel computing to all scientists and engineers. The students will be taught how to solve scientific problems using parallel computing technologies. They will be introduced to basic terminologies and concepts of parallel computing, learn how to use MPI to develop parallel programs, and study how to design and analyze parallel algorithms. Prerequisite: CSCI220 with a grade of C- or higher or CSCI262 with a grade of C- or higher, CSCI341.

CSCI441. COMPUTER GRAPHICS. 3.0 Semester Hrs.

Equivalent with MATH441,

(I) This class focuses on the basic 3D rendering and modeling techniques. In particular, it covers the graphics pipeline, elements of global illumination, modeling techniques based on polynomial curves and patches, and shader programming using the GPU. Prerequisite: CSCI220 with a grade of C- or higher or CSCI262 with a grade of C- or higher, MATH332.

CSCI442. OPERATING SYSTEMS. 3.0 Semester Hrs.

(I, II) Introduces the essential concepts in the design and implementation of operating systems: what they can do, what they contain, and how they are implemented. Despite rapid OS growth and development, the fundamental concepts learned in this course will endure. We will cover the following high-level OS topics, roughly in this order: computer systems, processes, processor scheduling, memory management, virtual memory, threads, and process/thread synchronization. This course provides insight into the internal structure of operating systems; emphasis is on concepts and techniques that are valid for all computers. Prerequisite: CSCI220 with a grade of C- or higher or CSCI262 with a grade of C- or higher, CSCI274, CSCI341.

CSCI443. ADVANCED PROGRAMMING CONCEPTS USING JAVA. 3.0 Semester Hrs.

(I, II) This course will quickly review programming constructs using the syntax and semantics of the Java programming language. It will compare the constructs of Java with other languages and discuss program design and implementation. Object oriented programming concepts will be reviewed and applications, applets, servlets, graphical user interfaces, threading, exception handling, JDBC, and network - ing as implemented in Java will be discussed. The basics of the Java Virtual Machine will be presented. 3 hours lecture, 3 semester hours Prerequisite: CSCI306.

CSCI444. ADVANCED COMPUTER GRAPHICS. 3.0 Semester Hrs.

Equivalent with MATH444,

(II) This is an advanced computer graphics course, focusing on modern rendering and geometric modeling techniques. Students will learn a variety of mathematical and algorithmic techniques that can be used to develop high-quality computer graphic software. Runtime performance will be evaluated to create optimized real-time graphics applications. In particular, the course will cover global illumination, GPU programming, and virtual and augmented reality. Prerequisites: CSCI441. 3 hours lecture; 3 semester hours.

CSCI445. WEB PROGRAMMING. 3.0 Semester Hrs.

Web Programming is a course for programmers who want to develop web-based applications. It covers basic website design extended by client-side and server-side programming. Students should acquire an understanding of the role and application of web standards to website development. Topics include Cascading Style Sheets (CSS), JavaScript, PHP and database connectivity. At the conclusion of the course students should feel confident that they can design and develop dynamic Web applications on their own. Prerequisite: CSCI306. Co-requisite: CSCI403.

CSCI446. WEB APPLICATIONS. 3.0 Semester Hrs.

(II) In Web Applications students will learn how to build effective web-based applications. At the completion of this course, students should know HTTP, Hypertext Markup Language (HTML), Cascading Style Sheets (CSS), JavaScript, Ajax, and RESTful architectures. Additionally students should have considered a variety of issues related to web application architecture, including but not limited to security, performance, web frameworks and cloud-based deployment environments. 3 hours lecture; 3 semester hours. Prerequisite: CSCI220 or CSCI262. Co-requisite: CSCI403.

CSCI448. MOBILE APPLICATION DEVELOPMENT. 3.0 Semester Hrs.

(II) This course covers basic and advanced topics in mobile application development. Topics include the mobile application lifecycle, user interface components and layouts, storing persistent data, accessing network resources, using location and sensor APIs including GPS and accelerometer, starting and stopping system services, and threading. This is a project-based course where students will design and develop complete applications. Prerequisite: CSCI306 with a grade of C- or higher. 3 hours lecture; 3 semester hours.

CSCI455. GAME THEORY AND NETWORKS. 3.0 Semester Hrs.

Equivalent with CSCI555,

(I) An introduction to fundamental concepts of game theory with a focus on the applications in networks. Game theory is the study that analyzes the strategic interactions among autonomous decision-makers. Originated from economics. Influenced many areas in Computer Science, including artificial intelligence, e-commerce, theory, and security and privacy. Provides tools and knowledge for modeling and analyzing real-world problems. Prerequisites: CSCI358, CSCI406. 3 hours lecture; 3 semester hours.

CSCI470. INTRODUCTION TO MACHINE LEARNING. 3.0 Semester Hrs.

(I) The goal of machine learning is to build computer systems that improve automatically with experience, which has been successfully applied to a variety of application areas, including, for example, gene discovery, financial forecasting, and credit card fraud detection. This introductory course will study both the theoretical properties of machine learning algorithms and their practical applications. Students will have an opportunity to experiment with machine learning techniques and apply them to a selected problem in the context of term projects. Prerequisite: CSCI101 or CSCI102 or CSCI200 or CSCI261, MATH201, MATH332.

CSCI471. COMPUTER NETWORKS I. 3.0 Semester Hrs.

This introduction to computer networks covers the fundamentals of computer communications, using TCP/IP standardized protocols as the main case study. The application layer and transport layer of communication protocols will be covered in depth. Detailed topics include application layer protocols (HTTP, FTP, SMTP, and DNS), transport layer protocols (reliable data transfer, connection management, and congestion control), network layer protocols, and link layer protocols. In addition, students will program client/server network applications. Prerequisite: (CSCI220 or CSCI262) AND (CSCI210 or CSCI274).

CSCI473. ROBOT PROGRAMMING AND PERCEPTION. 3.0 Semester Hrs.

Equivalent with CSCI573,

In this class students will learn the basics of integrated robot system programming and the design and use of algorithms for robot perception. Students will learn how to use the ROS robot middleware for the design of robot systems for perceiving and navigating the world; develop reinforcement learning based models for perception-informed autonomous navigation; and develop computational models for 3D robot perception and perceptual representation of human data. Prerequisite: (CSCI220 or CSCI262),and,(MATH201 or MATH334).

CSCI474. INTRODUCTION TO CRYPTOGRAPHY. 3.0 Semester Hrs.

Equivalent with MATH474,

(II) This course is primarily oriented towards the mathematical aspects of cryptography, but is also closely related to practical and theoretical issues of computer security. The course provides mathematical background required for cryptography, including relevant aspects of number theory and mathematical statistics. The following aspects of cryptography will be covered: symmetric and asymmetric encryption, computational number theory, quantum encryption, RSA and discrete log systems, SHA, steganography, chaotic and pseudo-random sequences, message authentication, digital signatures, key distribution and key management, and block ciphers. Many practical approaches and most commonly used techniques will be considered and illustrated with real-life examples. Prerequisite: CSCI220 or CSCI262, CSCI358, MATH334 or MATH335 or MATH201.

CSCI475. INFORMATION SECURITY AND PRIVACY. 3.0 Semester Hrs.

(I) Information Security and Privacy provides a hands-on introduction to the principles and best practices in information and computer security. Lecture topics will include basic components of information security including threat assessment and mitigation, policy development, forensics investigation, and the legal and political dimensions of information security. Completion of CSCI274 recommended. Prerequisite: CSCI220 or CSCI262, CSCI 341, CSCI274.

CSCI477. ELEMENTS OF GAMES AND GAME DEVELOPMENT. 3.0 Semester Hrs.

This course provides an overview of computer and video game development along with practical game projects designed to introduce the student to the computer entertainment industry. Topics will include the nature of games, the game player, game play, game design, game mechanics, story and character, game worlds, interface and the game development process. Students will be required to develop code both in C++ and with the use of a game engine. Prerequisite: CSCI220 or CSCI262.

CSCI478. INTRODUCTION TO BIOINFORMATICS. 3.0 Semester Hrs.

Bioinformatics is the theory, application and development of computing tools to solve problems and create hypotheses in all areas of biological sciences, which has contributed to advances in biology by providing tools that handle datasets too large and/or complex for manual analysis. This course focuses on an introduction to computational analysis of genetic variation and computational interdisciplinary research in genetics. The topics of this course include introduction to genetics, identification of genes involved in disease, inferring human population history, technologies for obtaining genetic information, and genetic sequencing, with an emphasis on formulating interdisciplinary problems as computational problems and then solving those problems using computational techniques from statistics and computer science. Prerequisite: CSCI101 or CSCI102 or CSCI128 or CSCI200 or CSCI261.

CSCI480. COMPUTER SCIENCE HONORS THESIS. 3.0 Semester Hrs.

(I, II) Prerequisite: CSCI306. 3 hours research; 3 semester hours. Repeatable for credit up to 6 semester hours.

CSCI490. PROGRAMMING CHALLENGES III. 1.0 Semester Hr.

This course is the third of three courses in the Programming Challenges sequence which covers problem solving patterns and paradigms found in technical interviews and programming competitions. The students will learn more advanced dynamic programming, graph theory, and algorithms while focusing on algorithmic complexity to solve problems in a finite amount of time. Prerequisite: CSCI390. Co-requisite: CSCI406.

CSCI498. SPECIAL TOPICS. 1-6 Semester Hr.

(I, II) Pilot course or special topics course. Topics chosen from special interests of instructor(s) and student(s). Usually the course is offered only once. Prerequisite: none. Variable credit; 1 to 6 credit hours. Repeatable for credit under different titles.

CSCI499. INDEPENDENT STUDY. 1-6 Semester Hr.

(I, II) Individual research or special problem projects supervised by a faculty member, when a student and instructor agree on a subject matter, content, and credit hours. Prerequisite: "Independent Study" form must be completed and submitted to the Registrar. Variable credit; 1 to 6 credit hours. Repeatable for credit.

Professor and Department Head

Iris Bahar

Professors

Qi Han

Dinesh Mehta

Associate Professors

Hua Wang

Bo Wu

Thomas Williams

Dejun Yang

Chuan Yue

Assistant Professors

Mehmet Belviranli

Dong Chen

Neil Dantam

Gabriel Fierro

C. Estelle Smith

Teaching Professors

Tolga Can

Vibhuti Dave

Jeffrey Paone

Christopher Painter-Wakefield

Teaching Associate Professors

Wendy Fisher, Assistant Department Head

Kathleen Kelly

Teaching Assistant Professors

Amelia Read

Robert Thompson

Zibo Wang

Professors of Practice

Mark Baldwin

Phil Romig

Visiting Teaching Associate Professor

Keith Hellman

Emeriti Professors

Tracy Camp, Emeritus Professor

William Hoff, Emeritus Associate Professor

Cyndi Rader, Emeritus Teaching Professor