

CICS 191FY1 CICS First Year Seminar

INSTRUCTOR(S): STAFF, Simon Andrews, Md Abdul Aowal, Colette Basiliere, Alyxander Burns, Casey Hayman, William Lee, Marc Liberatore, Laura Melbin, Cooper Siarist, Pamela Stawasz, Zachary White

An exploration of computer science for first-year computer science majors and exploratory track students, focusing on a single topic. 1 credit. REGISTRATION AVAILABLE ONLY DURING SUMMER ORIENTATION.

Sections 01 and 15- Pathways in Computer Science

Sections 02 and 06- Know your impact: Ethics and Accessibility

Section 03 and 18- Thinking with Machine Learning

Sections 04, 07, 14, and 16- Math Puzzles

Section 05- History of Computer Science

Sections 08 and 11 - Can Data Speak for Themselves?

Section 19 - Study Skills RAP

Sections 21 and 22 - Data Science for Good

CICS 256 Make: A Hands-on Introduction to Physical Computing

INSTRUCTOR(S): Tauhidur Rahman

Inspired by the Maker movement, this course provides a hands-on introduction to physical computing: sensing and responding to the physical world using computers. Specific topics include: basic electronics and circuit design, microcontroller programming using Arduinos, sensing and responding to the physical world, rapid prototyping (3D printing and laser cutting etc.), soft circuits and wearable electronics. The course will encourage and empower students to invent, design, and build practical hardware projects that interact with the physical world. This course has a required lab section, and counts as one of the CS Lab Science Requirement courses for the BS-CS. Prerequisite: COMPSCI 187 and Basic Math Skills (R1). 4 credits.

CICS 290A Introduction to Research in the Discipline

INSTRUCTOR(S): Neena Thota

The 'Introduction to Research in the Discipline' course is part of the CICS Early Research Scholars Program (ERSP). It provides a group-based, dual-mentored research structure designed to provide a supportive and inclusive first research experience for a large number of early-career Computer Science and Informatics majors. 1 credit.

CICS 291T Seminar - CICS Transfer Success

INSTRUCTOR(S): Jason Valladares

This seminar is intended to help you become fully prepared to succeed in CICS at UMass. Students in this seminar will be led by an instructor with a detailed understanding of the transfer student experience, and supported by various staff members in CICS. You will learn about which campus and College resources will be most helpful to you, how to best utilize these resources, and where you can look for other opportunities to connect. 1 credit.

CICS 298A Practicum - Leadership: Communicating Across Expertise

INSTRUCTOR(S): Emma Anderson

No matter where you end up in tech, you will need to explain concepts, products and ideas to people with different technical backgrounds. This course is intended to help prepare you for these communication tasks. Through the lens of tutoring, we will work on explaining technical ideas clearly and compassionately to others. We will do some theoretical study, including a history of CS education as well as brain and learning science, and some practice, including tutoring beginning students in CS. This course is intended for a broad range of students looking to pursue careers in tech, but will be particularly useful for those who are currently UCAs or intending to apply for UCA positions in the future. Prerequisite: COMPSCI 186 or 187. 1 credit.

CICS 305 Social Issues in Computing

INSTRUCTOR(S): Siobhan Mei, Justin Obara

Through a careful analysis and discussion of a range of computing issues, topics, and policies, we will explore various impacts of computers on modern society. This class satisfies the Junior Year Writing requirement by providing directed practice and specific instruction in a range of writing genres. Students will produce approximately 20-25 pages of polished written work over the course of the semester. Prerequisite: ENGLWRIT 112 (or English Writing waiver), COMPSCI 220, COMPSCI 230 and COMPSCI 240 (or 250), OR ENGLWRIT 112 (or English Writing waiver) AND INFO 248. 3 credits.

CICS 397A Special Topics - Predictive Analytics with Python

INSTRUCTOR(S): Matthew Rattigan

Modern technological advances are generating ever-greater volumes of data, and there are an astounding number of opportunities to use these data sets for good (and bad) in the applied sciences, business, social media, politics, cyber security, etc. The goal of this course is to familiarize participants with some of the most commonly used data analytics techniques, including methods for reducing data to informative statistics, predictive modeling, and cluster analysis. Students in this course will learn and use the Python programming language, creating scripts from the ground up to collect, manipulate, and analyze real and fascinating data sets. We will learn to ask and answer questions from data, and will cover all phases of the analytics process, from basic data wrangling and transformation to communicating through visualization. This course requires mathematical background in probability and statistics, and experience programming in some modern programming language. Does not count as a CS Elective (BA or BS). Prerequisites: Either COMPSCI 119 (or 121 or 186 or 187) and STATISTCS 240 (or OIM 240 or PSYCH 240 or STATISTCS 515 or RES-ECON 212 or SOCIOL 212). 3 credits.

CICS 580 Introduction to Numerical Computing with Python

INSTRUCTOR(S): Justin Payan

This course is an introduction to computer programming for numerical computing. The course is based on the computer programming language Python and is suitable for students with no programming or numerical computing background who are interested in taking courses in machine learning, natural language processing, or data science. The course will cover fundamental programming, numerical computing, and numerical linear algebra topics, along with the Python libraries that implement the corresponding data structures and algorithms. The course will include hands-on programming assignments and a project. No prior programming experience is required. Familiarity with undergraduate-level probability, statistics and linear algebra is assumed. Does not count toward graduate degrees. Open to Graduate students only. 1 credit.

COMPSCI 119 Introduction to Programming

INSTRUCTOR(S): William Verts

A complete introduction to computer programming using the Python language. Topics include coverage of all the supported data types and program code structures, functions (up through lambda expressions and recursion), reasoning about and debugging existing code, implementation of custom libraries, selection of data structures, and the fundamentals of object-oriented programming. Students will create, debug, and run Python 3 programs that explore each of these topics in turn, from simple loops up through the processing of large data sets, and eventually to the creation of professional-quality libraries to synthesize graphics images and audio files. No prior programming experience expected. Not open to Computer Science majors. 3 credits.

COMPSCI 121 Introduction to Problem Solving with Computers

INSTRUCTOR(S): Eduardo Calle Ortiz, Meng-Chieh Chiu, Jaime Davila, Collin Giguere, Peter Klemperer, Adam Kohan, Spencer Lane, Ghazaleh Parvini, Cheryl Swanier

COMPSCI 121 provides an introduction to problem solving and computer programming using the programming language Java. The course teaches how real-world problems can be solved computationally using the object-oriented metaphor that underlies Java. Concepts and techniques covered include data types, expressions, objects, methods, top-down program design, program testing and debugging, state representation, interactive programs, data abstraction, conditionals, iteration, interfaces, inheritance, polymorphism, arrays, graphics, and GUIs. No previous programming experience is required; however, this course is intended for Computer Science majors or those who plan on applying to the major. Non-majors are strongly encouraged to take one of our programming courses designed for non-majors. Use of a laptop computer on which you can install software is required. Prerequisite: R1 (or a score of 20 or higher on the math placement test Part A), or one of the following courses: MATH 101&102 or MATH 104 or MATH 127 or MATH 128 or MATH 131 or MATH 132. 4 credits.

COMPSCI 186 Using Data Structures

INSTRUCTOR(S): Peter Klemperer

COMPSCI 186 introduces foundational abstract data types and algorithms. The main focus is on the use of data structures in designing and developing programs to solve problems in a variety of domains. Specific topics include lists, sets, maps, graphs, stacks, queues, searching, and sorting. There will be weekly programming assignments, programming and written exercises in discussion sections, regular quizzes, and a cumulative final exam. This course is not a substitute for COMPSCI 187. If unsure of whether this course or COMPSCI 187 is more appropriate, contact instructor. Prerequisites: COMPSCI 121 and Basic Math Skills R1 (or a score of 20 or higher on the math placement test Part A), or one of the following courses: MATH 101&102 or MATH 104 or MATH 127 or MATH 128 or MATH 131 or MATH 132). 4 credits.

COMPSCI 187 Programming with Data Structures

INSTRUCTOR(S): Gordon Anderson, Neena Thota

The course introduces and develops methods for designing and implementing abstract data types using the Java programming language. The main focus is on how to build and encapsulate data objects and their associated operations. Specific topics include linked structures, recursive structures and algorithms, binary trees, balanced trees, and hash tables. These topics are fundamental to programming and are essential to other courses in computer science. The course involves weekly programming assignments, in-class quizzes, discussion section exercises, and multiple exams. Prerequisites: COMPSCI 121 (or equivalent Java experience). A grade of B or better in COMPSCI 121 (or a grade of C or better in COMPSCI 186 (or COMPSCI 190D) is required for students enrolling in COMPSCI 187 and Basic Math Skills (R1). Basic Java language concepts are introduced quickly; if unsure of background, contact instructor. 4 credits.

COMPSCI 190F Foundations of Data Science

INSTRUCTOR(S): Thomas Bernardin

The field of Data Science encompasses methods, processes, and systems that enable the extraction of useful knowledge from data. Foundations of Data Science introduces core data science concepts including computational and inferential thinking, along with core data science skills including computer programming and statistical methods. The course presents these topics in the context of hands-on analysis of real-world data sets, including economic data, document collections, geographical data, and social networks. The course also explores social issues surrounding data analysis such as privacy and design. Prerequisite: Completion of the R1 General Education Requirement (or a score of 20 or higher on the Math Placement Exam, Part A) or one of the following courses: Math 101 & 102, Math 104, 127, 128, 131, or 132. 4 credits.

COMPSCI 198C Practicum - Introduction to the C Programming Language

INSTRUCTOR(S): Meng-Chieh Chiu, J Moss, Timothy Richards

This practicum assumes general background and experience in computer programming (such as that provided by COMPSCI 121 or a similar introductory programming course) and some knowledge of data structures (such as that provided by COMPSCI 187). Content will include basic C data types, declarations, expressions, statements, and functions; simple use of macros; some common library calls (such as formatted input/output); basic pointer manipulation using linked lists; and introduction to using standard tools (gcc and make). 1 credit.

COMPSCI 220 Programming Methodology

INSTRUCTOR(S): Marius Minea, Emily Pruc

Development of individual skills necessary for designing, implementing, testing and modifying larger programs, including: use of integrated design environments, design strategies and patterns, testing, working with large code bases and libraries, code refactoring, and use of debuggers and tools for version control. There will be significant programming and a mid-term and final examination. Prerequisite: COMPSCI 187. 4 credits.

COMPSCI 230 Computer Systems Principles

INSTRUCTOR(S): Meng-Chieh Chiu

Large-scale software systems like Google - deployed over a world-wide network of hundreds of thousands of computers - have become a part of our lives. These are systems success stories - they are reliable, available ("up" nearly all the time), handle an unbelievable amount of load from users around the world, yet provide virtually instantaneous results. On the other hand, many computer systems don't perform nearly as well as Google - hence the now-cliché "the system is down." In this class, we study the scientific principles behind the construction of high-performance, scalable systems. The course begins with a discussion of C language, and moves up the stack from there to the features of modern architectures, assembly languages, and operating system services such as I/O and synchronization. Prerequisite: COMPSCI 187. 4 credits.

COMPSCI 240 Reasoning Under Uncertainty

INSTRUCTOR(S): Peter Haas, Jie Xiong

Development of mathematical reasoning skills for problems that involve uncertainty. Each concept will be illustrated by real-world examples and demonstrated through in-class and homework exercises. Counting and probability -- basic counting problems, probability definitions, mean, variance, binomial distribution, discrete random variables, continuous random variables, Markov and Chebyshev bounds, Laws of large number, and central limit theorem. Probabilistic reasoning -- conditional probability and odds, Bayes' Law, Markov Chains, Bayesian Network, Markov Decision Processes. Prerequisites: COMPSCI 187 and MATH 132. 4 credits.

COMPSCI 250 Introduction to Computation

INSTRUCTOR(S): David Barrington, Ghazaleh Parvini

Basic concepts of discrete mathematics useful to computer science: set theory, strings and formal languages, propositional and predicate calculus, relations and functions, basic number theory. Induction and recursion: interplay of inductive definition, inductive proof, and recursive algorithms. Graphs, trees, and search. Finite-state machines, regular languages, nondeterministic finite automata, Kleene's Theorem. Problem sets, 2 midterm exams, timed final. Prerequisite: COMPSCI 187 (or ECE 241) and MATH 132. 4 credits.

COMPSCI 311 Introduction to Algorithms

INSTRUCTOR(S): David Barrington, Daniel Sheldon, Ramesh Sitaraman

This course will introduce you to algorithms in a variety of areas of interest, such as sorting, searching, string-processing, and graph algorithms. You will learn to study the performance of various algorithms within a formal, mathematical framework. You will also learn how to design very efficient algorithms for many kinds of problems. There will be one or more programming assignments as well to help you relate the empirical performance of an algorithm to theoretical predictions. Mathematical experience (as provided by COMPSCI 250) is required. You should also be able to program in Java, C, or some other closely related language. Prerequisite: COMPSCI 187 and either COMPSCI 250 or MATH 455. 4 credits.

COMPSCI 320 Introduction to Software Engineering

INSTRUCTOR(S): Gordon Anderson, David Fisher

In this course, students learn and gain practical experience with software engineering principles and techniques. The practical experience centers on a semester-long team project in which a software development project is carried through all the stages of the software life cycle. Topics in this course include requirements analysis, specification, design, abstraction, programming style, testing, maintenance, communication, teamwork, and software project management. Particular emphasis is placed on communication and negotiation skills and on designing and developing maintainable software. Use of computer required. Several written assignments, in-class presentations, and a term project. This course satisfies the IE Requirement. Prerequisite: COMPSCI 220. 4 credits.

COMPSCI 325 Introduction to Human Computer Interaction

INSTRUCTOR(S): Mahmood Jasim

Human-Computer Interaction design is "design for human use". Computers are a ubiquitous part of many interactions in our lives, from the mundane everydayness of light switches and "smart" vending machines to entertainment and education to sophisticated instruments and complex energy and defense systems. In this course, we will challenge you to broaden your grasp of what a user interface can and should be, and try your hand at doing better yourself. It is a fast-paced, hands-on, project-based experience that will challenge many of your ideas of what computer science is and can be. It is designed around active lecture sessions supported by readings, working classes, and team projects, where students practice and explore the concepts introduced in lecture, and go well beyond them to learn and apply HCI techniques that build into group projects. More specifically, the course adopts a human-centered design (HCD) approach and teaches a highly iterative process called design thinking. The design thinking process draws heavily on the fundamentals of human-computer interaction (HCI) methods. I also cover design methodologies, evaluation methodologies (both quantitative and qualitative), human information processing, cognition, and perception. Prerequisite: COMPSCI 186 (or COMPSCI 187). 3 credits.

COMPSCI 326 Web Programming

INSTRUCTOR(S): Emery Berger

The web is arguably today's most important application platform. Web browsers run on practically every device, and even many phone applications are in fact web applications under the covers. This course will cover a broad range of client-side web technologies, including HTTP itself, HTML5, CSS, and JavaScript; it will additionally cover key concepts for the server side of web applications, including key value stores and SQL servers. This course will also cover key concepts and technologies including AJAX, JavaScript libraries (e.g., jQuery), and web security. This course is hands-on and heavily project-based; students will construct a substantial dynamic web application based on the concepts, technologies, and techniques presented during lectures and in readings. This course satisfies the IE Requirement. Prerequisites: COMPSCI 220 (OR COMPSCI 230), OR INFO 248 AND COMPSCI 186 (OR COMPSCI 187). Note: as the name 'web programming' denotes, programming is a key component of this class. Previous background in JavaScript is strongly recommended. 4 credits.

COMPSCI 328 Mobile Health Sensing and Analytics

INSTRUCTOR(S): Deepak Ganesan

In recent years, the ability to continuously monitor activities, health, and lifestyles of individuals using sensor technologies has reached unprecedented levels. The typical smartphone comes equipped with a plethora of sensors for monitoring activity, speech patterns, social interactions, and location. In addition, mobile accessories such as wearable wristbands and chestbands now enable routine and continuous monitoring of a host of physiological signals (e.g., heart rate, respiratory rate, skin Conductance, and others.). In conjunction, these sensors can enable higher-order inferences about more complex human activities/behavioral states (e.g., activity patterns, stress, sleep, etc.). Such ubiquitous sensing in daily life, referred to as mobile health sensing and monitoring, promises to revolutionize our understanding of human activities and health conditions. This course is an introduction to personal health sensing and monitoring through mobile phones and on-body sensors and addresses several aspects including mobile devices and applications for health, sensor data quality and reliability challenges, inference of key health assessments from sensor data including such as activity patterns, sleep patterns, or stress, sensor data visualization and feedback, and practical considerations such as battery lifetime. Prerequisite: COMPSCI 187 (OR INFO 248 AND COMPSCI 186). 3 credits.

COMPSCI 345 Practice and Applications of Data Management

INSTRUCTOR(S): Jaime Davila, Marco Serafini

Computing has become data-driven, and databases are now at the heart of commercial applications. The purpose of this course is to provide a comprehensive introduction to the use of data management systems within the context of various applications. Some of the covered topics include application-driven database design, schema refinement, implementation of basic transactions, data on the web, and data visualization. The class will follow a flipped classroom model; students will be required to review materials in preparation for each week and they will work collaboratively on practical problems in class. This course counts as a CS Elective toward the COMPSCI major (BA/BS). Students who have completed COMPSCI 445 are not eligible to take this course without instructor permission. Prerequisite: COMPSCI 187 (OR INFO 248 AND COMPSCI 186). 3 credits.

COMPSCI 377 Operating Systems

INSTRUCTOR(S): Mark Corner

In this course we examine the important problems in operating system design and implementation. The operating system provides a well-known, convenient, and efficient interface between user programs and the bare hardware of the computer on which they run. The operating system is responsible for allowing resources (e.g., disks, networks, and processors) to be shared, providing common services needed by many different programs (e.g., file service, the ability to start or stop processes, and access to the printer), and protecting individual programs from one another. The course will start with a brief historical perspective of the evolution of operating systems over the last fifty years, and then cover the major components of most operating systems. This discussion will cover the tradeoffs that can be made between performance and functionality during the design and implementation of an operating system. Particular emphasis will be given to three major OS subsystems: process management (processes, threads, CPU scheduling, synchronization, and deadlock), memory management (segmentation, paging, swapping), file systems, and operating system support for distributed systems. Prerequisites: COMPSCI 230. 4 credits.

COMPSCI 383 Artificial Intelligence

INSTRUCTOR(S): Matthew Rattigan

The course explores key concepts underlying intelligent systems, which are increasingly deployed in consumer products and online services. Topics include problem solving, state-space representation, heuristic search techniques, game playing, knowledge representation, logical reasoning, automated planning, reasoning under uncertainty, decision theory and machine learning. We will examine the use of these concepts in the design of intelligent agents in the context of several applications. Prerequisites: COMPSCI 220 (or COMPSCI 230) and COMPSCI 240 (or STAT 515). 3 credits.

COMPSCI 391L Seminar - Computer Crime Law and the Technologies of Investigation and Privacy

INSTRUCTOR(S): Marvin Cable

A study, analysis, and discussion of the legal issues related to crimes involving computers and networks, including topical actions by dissidents and governments. We will also study the technologies of forensic investigation, intelligence gathering, privacy enhancement, and censorship resistance. Our main legal topics will include recent and important case law, statutes, and constitutional clauses concerning authorization, access, search and seizure, wiretaps, the right to privacy, and FISA. Our technology topics will include methods of investigation and resistance in the context of the Internet and Cellular networks. Students are assumed to have no background in legal concepts. Students will be required to complete substantial legal readings, complete significant written analysis of rulings, learn about technologies in detail, and participate in lively class discussion. Prerequisite: COMPSCI 230 and ENGLWRIT 112. This course counts as a CS Elective toward the CS major. 3 credits.

COMPSCI 403 Introduction to Robotics: Perception, Mechanics, Dynamics, and Control

INSTRUCTOR(S): Donghyun Kim

This course covers basic methods and concepts in order to explain how robots work. We will study how they sense things in the world, how you make a robot move, and how robots can make their own decisions. We will study mechanisms (kinematics and dynamics), actuators, sensors, signal processing (with an emphasis on computer vision), feedback control theory, machine learning, and path planning. Students will build software systems for simulated robots to reinforce the material presented in class. Prerequisites: MATH 235 and COMPSCI 220 (or COMPSCI 230). 3 credits.

COMPSCI 446 Search Engines

INSTRUCTOR(S): David Fisher

This course provides an overview of the important issues in information retrieval, and how those issues affect the design and implementation of search engines. The course emphasizes the technology used in Web search engines, and the information retrieval theories and concepts that underlie all search applications. Mathematical experience (as provided by COMPSCI 240) is required. You should also be able to program in Java (or some other closely related language). Prerequisite: COMPSCI 240 or COMPSCI 383. 3 credits.

COMPSCI 453 Computer Networks

INSTRUCTOR(S): Parviz Kermani

Introduction to computer communication networks and protocols. Fundamental concepts in the design and analysis of computer networks. Topics include: layered network architectures, networked applications, network programming interfaces, transport, congestion, routing, data link protocols, local area and data center networks, network security, and wireless networks. Examples drawn from the Internet (e.g., TCP, UDP, and IP) protocol suite. Homework assignments involve programming and written tasks. In Fall 2020, this course will be taught as a primarily online class, without in-class lectures, but with extensive on-line material including recorded video material, interactive exercises, and online discussion. There will be an optional weekly, one-hour in-class and on-line open discussion period. Prerequisites: Experience programming; COMPSCI 230 or COMPSCI 377. 3 credits.

COMPSCI 460 Introduction to Computer and Network Security

INSTRUCTOR(S): Christopher Misra

This course provides an introduction to the principles and practice of computer and network security with a focus on both fundamentals and practical information. The three key topics of this course are cryptography, privacy, and network security. Subtopics include ciphers, key exchange, security services (integrity, availability, confidentiality, etc.), network and web based security attacks, anonymous communications, vulnerabilities, and countermeasures. Students will complete several lab assignments. Grades will be determined by class participation, lab work, homework, quizzes, and exams. Prerequisites include COMPSCI 377 and a familiarity with Unix. 3 credits.

COMPSCI 461 Secure Distributed Systems

INSTRUCTOR(S): Gregory Stone

This is a class devoted to the study of securing distributed systems, with blockchain-based cryptocurrencies serving as our real platform of interest. We'll start with the fundamentals of Lamport's, Fischer's, and Douceur's results that fence-in all consensus system, and discuss Byzantine fault tolerance. We'll also look at the efficiency of the network architectures for peer-to-peer/distributed system communication and attacks on their security, such as denial of service attacks. And we'll review relevant applied cryptography such as elliptic curves. We'll discuss in detail the mechanisms of Bitcoin and Ethereum and we'll program distributed applications for Ethereum. Other topics include economics and finance. Assignments will include programming projects and reading research papers. The grade is also based on exams and participation in discussion. The course is based on a flipped classroom and uses a hybrid instruction model. Some of the course content is delivered online, however students are required to attend weekly class meetings. This course counts as a CS Elective toward the CS major (BA/BS), as well as an Any 2 menu choice for the Security & Privacy track. Prerequisites: COMPSCI 326, COMPSCI 345, COMPSCI 377, COMPSCI 453, or COMPSCI 497P. 3 credits.

COMPSCI 490A Applications of Natural Language Processing

INSTRUCTOR(S): Brendan O'Connor, Laure Thompson

This course will introduce NLP methods and applications, such as text classification, sentiment analysis, machine translation, and other applications to identify and use the meaning of text. During the course, students will (1) learn fundamental methods and algorithms for NLP; (2) become familiar with key facts about human language that motivate them, and help practitioners know what problems are possible to solve; and (3) complete a series of hands-on projects to use, implement, experiment with, and improve NLP tools. This course counts as a CS Elective for the CS Major (BA or BS). Prerequisites: COMPSCI 220 and COMPSCI 240. An alternate prerequisite of LINGUIST 492B is acceptable for Linguistics majors. 3 credits.

COMPSCI 497S Special Topics - Scalable Web Systems

INSTRUCTOR(S): Timothy Richards

The web has become a large and complex area for application development. Access to an abundance of open source languages, libraries, and frameworks has led to the quick and easy construction of a variety of applications with several moving parts working in coordination to present to the user the illusion of a single program. In reality, web applications are extremely difficult to get right. They involve a large collection of coordinated services, multiple databases, complicated user interfaces, security and performance issues, and ever changing 3rd party services, spread across physical and virtual machines. These complications are further stressed by the large number of concurrent users that access these applications every second. This course will investigate several well known web-based applications and the technology and software architecture used to scale these applications. We will also study a specific topic related to scalability in software design in the context of web application architecture. This course counts as a CS Elective toward the CS major (BA/BS) and as an Elective toward the INFORM major. Prerequisites: COMPSCI 326. 3 credits.

COMPSCI 513 Logic in Computer Science

INSTRUCTOR(S): Neil Immerman

Rigorous introduction to mathematical logic from an algorithmic perspective. Topics include: Propositional logic: Horn clause satisfiability and SAT solvers; First Order Logic: soundness and completeness of resolution, compactness theorem. We will use the Coq theorem prover and Datalog. Prerequisites: COMPSCI 250 and COMPSCI 311. 3 credits.

COMPSCI 514 Algorithms for Data Science

INSTRUCTOR(S): Andrew McGregor, Cameron Musco

With the advent of social networks, ubiquitous sensors, and large-scale computational science, data scientists must deal with data that is massive in size, arrives at blinding speeds, and often must be processed within interactive or quasi-interactive time frames. This course studies the mathematical foundations of big data processing, developing algorithms and learning how to analyze them. We explore methods for sampling, sketching, and distributed processing of large scale databases, graphs, and data streams for purposes of scalable statistical description, querying, pattern mining, and learning. Undergraduate Prerequisites: COMPSCI 240 (or STATISTCS 515) and COMPSCI 311 both with a grade of B+ or better, OR (COMPSCI 240 and STATISTCS 515 and COMPSCI 311 and MATH 233 and MATH 235, all with a C or better). 3 credits

COMPSCI 520 Theory and Practice of Software Engineering

INSTRUCTOR(S): Heather Conboy

Introduces students to the principal activities and state-of-the-art techniques involved in developing high-quality software systems. Topics include: requirements engineering, formal specification methods, design principles & patterns, verification & validation, debugging, and automated software engineering. Undergraduate Prerequisites: COMPSCI 320. 3 credits.

COMPSCI 529 Software Engineering Project Management

INSTRUCTOR(S): Gordon Anderson, David Fisher

The purpose of this course is to provide students with practical experience in the management of software development projects. Students in this course will gain this experience by serving as software development team technical managers for teams of software engineering students in COMPSCI 320. As project managers, the students in COMPSCI 529 will be responsible for: supervising and managing the work of teams of COMPSCI 320 students; interfacing with the other COMPSCI 529 students managing other teams in the course; interfacing with the course instructor, course TA, and course customer. COMPSCI 529 students will be assigned readings in software engineering project management to provide a theoretical basis for their work in this course. But the majority of work in the course will be related to the actual management of assigned development teams. As team managers, COMPSCI 529 students will set goals and schedules for their teams, track and report team progress, negotiate with leaders of other teams and the course customer, and evaluate the work of members of their teams. COMPSCI 529 course assignments may include: written team goals, plans and schedules; periodic reports on team progress; documentation of agreements reached with other team leaders and customers; evaluations of the applicability of theoretical papers to the work of this course. This course will meet at the same times and places as COMPSCI 320. Additional meetings with team members and other students in COMPSCI 529 are also expected to be arranged by mutual agreement. An additional one hour weekly meeting of all of the students in COMPSCI 529 is required. Enrollment in this course is only by permission of the instructor, and is restricted to students who have previously taken COMPSCI 320, and received a grade of A or A-. 3 credits.

COMPSCI 532 Systems for Data Science

INSTRUCTOR(S): Hui Guan

In this course, students will learn the fundamentals behind large-scale systems in the context of data science. We will cover the issues involved in scaling up (to many processors) and out (to many nodes) parallelism in order to perform fast analyses on large datasets. These include locality and data representation, concurrency, distributed databases and systems, performance analysis and understanding. We will explore the details of existing and emerging data science platforms, including MapReduce-Hadoop, Spark, and more. Undergraduate Prerequisites: COMPSCI 311, COMPSCI 345, and COMPSCI 377. 3 credits.

COMPSCI 563 Internet Law and Policy

INSTRUCTOR(S): Marvin Cable

This course is meant for those looking for legal knowledge for use in computing- and Internet-related endeavors. The course will include topics related to security, policy, and the use of machine learning and related technologies. In addition, students will be assigned law review articles and will learn to do legal research so that they can remain updated after the course ends. Topics covered are all in the context of the ubiquity of the Internet and computing, and they include: basic legal principles, contract law, substantive laws, intellectual property law, ethics, dealing with third parties, policy issues, and topical issues such as implications of applying machine learning technology. This course was formerly numbered as INFOSEC 690L. 3 credits.

COMPSCI 576 Game Programming

INSTRUCTOR(S): Evangelos Kalogerakis

Game Programming introduces students to concepts of computer game development, including 2D and 3D modeling, character design, animation, game art, basic game AI, audio and video effects. The course will help students build the programming skills needed to turn ideas into games. Both runtime systems and the asset pipelines will be covered. Students will work on various game programming exercises with modern game engines and graphics APIs. This course counts as a CS Elective for the CS Major (BA or BS). Undergraduate Prerequisites: COMPSCI 220, 311, and MATH 235. 3 credits.

COMPSCI 589 Machine Learning

INSTRUCTOR(S): Justin Domke, Madalina Fiterau Brostean

This course will introduce core machine learning models and algorithms for classification, regression, clustering, and dimensionality reduction. On the theory side, the course will cover the mathematical foundations behind the most common machine learning algorithms. It will focus on understanding models and the relationships between them. On the applied side, the course will focus on effectively using machine learning methods to solve real-world problems with an emphasis on model selection, regularization, design of experiments, and presentation and interpretation of results.

The course will be held in a flipped-classroom manner, with students being assigned pre-recorded videos, and the lectures being reserved for discussions, including Q&A on the lecture topics, exercises, connecting the lecture abstractions to real-world application, implementation considerations and demos. The assignments will involve both mathematical problems and implementation tasks. Knowledge of a high-level programming language is absolutely necessary. Python is most commonly used, but languages such as Matlab, R, Scala, Julia would also be suitable. Strong foundations in linear algebra, calculus, probability and statistics are essential for the successful completion of this course. Undergraduate Prerequisites: COMPSCI 383 and MATH 235. 3 credits.

COMPSCI 590CC Cloud Computing

INSTRUCTOR(S): Arun Venkataramani

This course will teach the principles and practice of distributed systems as applied in today's cloud computing environments. The course will cover concepts in distributed computing including distributed clocks, consistency, fault tolerance, and consensus. The course will also cover popular cloud computing service models, MapReduce and related programming models, datacenter architectures, and security and privacy issues in public clouds. The course will expose students to popular cloud platforms such as Amazon EC2, Google AppEngine, and Microsoft Azure. The assignments in the course will include a nontrivial programming component that assumes familiarity with multithreaded programming and network programming. This course counts as a CS Elective toward the CS major (BA or BS). Prerequisite: COMPSCI 377 or an equivalent first course in operating systems. 3 credits.

COMPSCI 590ED Educational Data Mining and Learner Analytics

INSTRUCTOR(S): Ivon Arroyo, Shiting Lan

The primary goal of this course is for students to be comfortable with exploring educational datasets and analyzing them, finding patterns in educational data, creating models that summarize and describe data, and creating predictive models, using a variety of techniques that span machine learning, data mining, and statistics. We will work with datasets that come from educational learning technologies, tutoring and assessment software, as well as other datasets that students might be interested in (e.g. datasets from students grades classes, many of which are publicly available through national sources). Students are encouraged to bring their own datasets of interest, if they have them or want to. Another goal is to understand how current researchers in the field of educational data mining, AI in Education, and educational technology are using a variety of techniques to understand and model their data. We will cover many different methods, from machine learning methods applied to educational data and classic models inspired by cognitive theory. We will also cover various applications including computerized adaptive testing, affect detection, reading and writing analysis, activity log analysis, and instructional policy design. This course counts as a CS Elective toward the CS major (BA or BS). Cross-Listed with EDUC 590ED. Prerequisites: MATH 545 and COMPSCI 240 and STATS 515 C or better. MATH 545 can be skipped by students who have taken MATH 235 and MATH 233 both with B+ or better. STATS 515 can be skipped by students who have taken COMPSCI 240 with a B+ or better. 3 credits.

COMPSCI 590T Algorithmic Fairness and Strategic Behavior

INSTRUCTOR(S): Yair Zick

Recent years have seen a dramatic rise in the use of algorithms for solving problems involving strategic decision makers. Deployed algorithms now assist in a variety of economic interactions: assigning medical residents to schools, allocating students to courses, allocating security resources in airports, allocating computational resources and dividing rent. We will explore foundational topics at the intersection of economics and computation, starting with the foundations of game theory: Nash equilibria, the theory of cooperative games, before proceeding to covering more advanced topics: matching algorithms, allocation of indivisible goods, and mechanism design. This course counts as a CS Elective for the CS Major (BA or BS). Prerequisites: COMPSCI 240 and COMPSCI 250. 3 credits.

COMPSCI 597N Special Topics - Introduction to Computer and Network Security

INSTRUCTOR(S): Parviz Kermani

This course provides an introduction to the principles and practice of computer and network security with a focus on both fundamentals and practical information. The three key topics of this course are cryptography, privacy, and network security. Subtopics include ciphers, key exchange, security services (integrity, availability, confidentiality, etc.), network and web based security attacks, anonymous communications, vulnerabilities, and countermeasures. Students will complete several lab assignments. Grades will be determined by class participation, lab work, homework, quizzes, and exams. Open to Graduate students only. 3 credits.

COMPSCI 611 Advanced Algorithms

INSTRUCTOR(S): Hung Le

Principles underlying the design and analysis of efficient algorithms. Topics to be covered include: divide-and-conquer algorithms, graph algorithms, matroids and greedy algorithms, randomized algorithms, NP-completeness, approximation algorithms, linear programming. Prerequisites: The mathematical maturity expected of incoming Computer Science graduate students, knowledge of algorithms at the level of COMPSCI 311. 3 credits.

COMPSCI 613 Advanced Logic in Computer Science

INSTRUCTOR(S): Neil Immerman

Rigorous introduction to mathematical logic from an algorithmic perspective. Topics include: Propositional logic: Horn clause satisfiability and SAT solvers; First Order Logic: soundness and completeness of resolution, compactness theorem, automatic theorem proving, model checking. We will learn about and use the Coq theorem prover, Datalog, a Model Checker, and SAT and SMT solvers. Prerequisites: Students taking this course should have undergraduate preparation in discrete math and algorithms. Requirements will include readings, class participation, weekly problem sets, a midterm and a final project. 3 credits.

COMPSCI 646 Information Retrieval

INSTRUCTOR(S): Hamed Zamani

The course will cover basic and advanced techniques for text-based information systems. Topics covered include retrieval models, indexing and text representation, browsing and query reformulation, data-intensive computing approaches, evaluation, and issues surrounding implementation. The course will include a substantial project such as the implementation of major elements of search engines and applications. 3 credits.

COMPSCI 661 Secure Distributed Systems

INSTRUCTOR(S): Gregory Stone

This is a class devoted to the study of securing distributed systems, with blockchain-based cryptocurrencies serving as our real platform of interest. We'll start with the fundamentals of Lamport's, Fischer's, and Douceur's results that fence-in all consensus system, and discuss Byzantine fault tolerance. We'll also look at the efficiency of the network architectures for peer-to-peer/distributed system communication and attacks on their security, such as denial of service attacks. And we'll review relevant applied cryptography such as elliptic curves. We'll discuss in detail the mechanisms of Bitcoin and Ethereum and we'll program distributed applications for Ethereum. Other topics include economics and finance. Assignments will include programming projects and reading research papers. The grade is also based on exams and participation in discussion. The course is based on a "flipped classroom". Course was previously COMPSCI 690P. 3 credits.

COMPSCI 682 Neural Networks: A Modern Introduction

INSTRUCTOR(S): Erik Learned-Miller

This course will focus on modern, practical methods for deep learning with neural networks. The course will begin with a description of simple classifiers such as perceptrons and logistic regression classifiers, and move on to standard neural networks, convolutional neural networks, some elements of recurrent neural networks, and transformers. The emphasis will be on understanding the basics and on practical application more than on theory. Many applications will be in computer vision, but we will make an effort to cover some natural language processing (NLP) applications as well. The current plan is to use Python and associated packages such as Numpy and TensorFlow. Required background includes Linear Algebra, Probability and Statistics, and Multivariate Calculus. All assignments will be in the Python programming language. 3 credits.

COMPSCI 685 Advanced Natural Language Processing

INSTRUCTOR(S): Mohit Iyyer

This course covers a broad range of advanced level topics in natural language processing. It is intended for graduate students in computer science who have familiarity with machine learning fundamentals, and previous course or research experience in natural language processing. It may also be appropriate for computationally sophisticated students in linguistics and related areas. Topics include probabilistic models of language, computationally tractable linguistic representations for syntax and semantics, neural network models for language, and selected topics in discourse and text mining. After completing the course, students should be able to read and evaluate current NLP research papers. Coursework includes a research literature review, homework assignments, and a final project. 3 credits.

COMPSCI 687 Reinforcement Learning

INSTRUCTOR(S): Bruno Castro da Silva

This course will provide an introduction to, and comprehensive overview of, reinforcement learning. Reinforcement learning algorithms repeatedly answer the question "What should be done next?", and they can learn via trial-and-error to answer these questions even when there is no supervisor telling the algorithm what the correct answer would have been. Applications of reinforcement learning span across medicine (How much insulin should be injected next? What drug should be given next?), marketing (What ad should be shown next?), robotics (How much power should be given to the motor?), game playing (What move should be made next?), environmental applications (Which countermeasure for an invasive species should be deployed next?), and dialogue systems (What type of sentence should be spoken next?), among many others. Broad topics covered in this course will include: Markov decision processes, reinforcement learning algorithms (model-based/model-free, batch/online, value function-based, actor-critics, policy gradient methods, etc.), and representations for reinforcement learning. Special topics may include ensuring the safety of reinforcement learning algorithms, hierarchical reinforcement learning, theoretical reinforcement learning, multi-agent reinforcement learning, and connections to animal learning. This course will emphasize hands-on experience, and assignments will require the implementation and application of many of the algorithms discussed in class. A background in machine learning (COMPSCI 589 or 689) and artificial intelligence (COMPSCI 683) is assumed. Assignments may require familiarity with C++ and/or Python. 3 credits.

COMPSCI 689 Machine Learning

INSTRUCTOR(S): Benjamin Marlin

Machine learning is the computational study of artificial systems that can adapt to novel situations, discover patterns from data, and improve performance with practice. This course will cover the popular frameworks for learning, including supervised learning, reinforcement learning, and unsupervised learning. The course will provide a state-of-the-art overview of the field, emphasizing the core statistical foundations. Detailed course topics: overview of different learning frameworks such as supervised learning, reinforcement learning, and unsupervised learning; mathematical foundations of statistical estimation; maximum likelihood and maximum a posteriori (MAP) estimation; missing data and expectation maximization (EM); graphical models including mixture models, hidden-Markov models; logistic regression and generalized linear models; maximum entropy and undirected graphical models; nonparametric models including nearest neighbor methods and kernel-based methods; dimensionality reduction methods (PCA and LDA); computational learning theory and VC-dimension; reinforcement learning; state-of-the-art applications including bioinformatics, information retrieval, robotics, sensor networks and vision. Prerequisites: undergraduate level probability and statistics, linear algebra, calculus, AI; computer programming in some high level language. 3 credits.

COMPSCI 690Q Quantum Information Systems

INSTRUCTOR(S): Donald Towsley

Fundamentals of quantum information systems, including quantum computation, quantum cryptography, and quantum information theory. Topics include: quantum circuit model, qubits, unitary operators, measurement, entanglement, quantum algorithms for factoring and search, quantum key distribution, error-correction and fault-tolerance, information capacity of quantum channels, complexity of quantum computation. 3 credits.

COMPSCI 691DD Seminar - Research Methods in Empirical Computer Science

INSTRUCTOR(S): David Jensen

This course introduces graduate and undergraduate students to concepts, practices, and tools for conducting effective research. Students will learn basic methods for activities such as reading technical papers, selecting research topics, devising research questions, planning research, analyzing experimental results, modeling and simulation of computational phenomena, and synthesizing broader theories. The course is structured around three activities: lectures on basic concepts of research strategy and techniques, discussions of technical papers, and preparation and review of written assignments. Significant reading, reviewing, and writing is required. Students are expected to participate actively in class discussions and to provide meaningful comments on the work of other students. For PhD students, this course helps accelerate their current and future research. For MS students, this course provides a grounding in research methods that will aid entry into research-oriented industrial positions and provide information about continuing to PhD studies. For undergraduates considering graduate studies, this course will help inform and accelerate that direction. This course can be used to satisfy the 499Y requirement for Departmental and Multidisciplinary Honors students whose theses or projects have a substantial empirical component. Approval of the Computer Science Honors Program Director is required prior to registering. 3 credits.

COMPSCI 692C Seminar - Equitable and Explainable Artificial Intelligence

INSTRUCTOR(S): Przemyslaw Adam Grabowicz

This seminar will focus on recent research into equitable and transparent algorithms and systems. We will review cutting-edge research that supports the properties of fairness, accountability, and transparency across various research areas, in particular fair machine learning, explainable artificial intelligence, and their interdisciplinary underpinnings. The seminar will offer introductory lectures describing the origins of relevant research problems, highlighting major threads and approaches in this vivid research space, and describing the relations between them. The course will primarily involve reading and discussing papers and book chapters. 1 credit.

COMPSCI 692N Seminar - Novel Computational Hardware and Machine Learning

INSTRUCTOR(S): Edward Rietman

The seminar will consist of readings and discussions on novel computational systems and machine learning. Will include oscillatory computing, DNA computing, quantum computing, molecular computing, non-Turing architecture, among others. We will also look at novel machine learning algorithms. 3 credits.

COMPSCI 698W Practicum - CS Research Writing Practicum

INSTRUCTOR(S): Justin Obara

This CS research writing class uses a workshop format to focus on structure and phrasing while engaging students in a process-based approach to writing. Instruction will emphasize genre and discourse analysis and engage students in activities to strengthen audience awareness. As such, students will analyze representative examples of computer science research writing for stylistic and argumentative conventions and then integrate the awareness of these conventions and "moves" into their own writing. Students will produce or substantially revise a complete piece of writing. 6 weeks. 1 credit.

COMPSCI 701 Advanced Topics in Computer Science

INSTRUCTOR(S):

Advanced Topics in Computer Science Master's Project: Advanced research project in Computer Science. The 3 credit option is for the second semester of a two semester sequence, 701Y followed by 701. The 6 credit option is for a project that will be begun and completed within the same semester. 6 credits.

COMPSCI 701Y Advanced Topics in Computer Science (1st Semester)

INSTRUCTOR(S):

Advanced Topics in Computer Science Master's Project: Advanced research project in Computer Science. Indicates the first semester of a two-semester sequence, 701Y (3 credits) followed by 701 (3 credits), with grade for both assigned at the end. 3 credits.

COMPSCI 741 Complexity Theory

INSTRUCTOR(S): David Barrington

The study of the resources required to solve different problems in various abstract models of computation. Sequential computation: Turing machines, non-determinism, alternation, algebraic automata theory. Parallel computation: Boolean circuits, branching programs, uniformity. Lower bounds for circuit models. Descriptive Complexity. Possible optional topics depending on student interest: Approximation of NP-complete problems, interactive proofs, non-uniform finite automata, dynamic complexity. Prerequisite: COMPSCI 601 or permission of instructor. Intended for interested general graduate students as well as those concentrating in theory. Course format: Probably 2-3 problem sets, class presentation (may be survey or actual research, might lead to a synthesis paper). Textbook: Xeroxed notes and research papers. 3 credits.

COMPSCI 879 Teaching Assistants as Tomorrow's Faculty

INSTRUCTOR(S): Neena Thota

Teaching Assistants as Tomorrow's Faculty prepares Teaching Assistants (TAs) at the College of Information and Computer Sciences to fulfill their duties in an effective and pedagogically sound manner. The one credit (not repeatable) course is semester long and taken by all TAs prior to assuming assistantship. 1 credit.

COMPSCI 891M Seminar - Theory of Computation

INSTRUCTOR(S): Cameron Musco

The theory seminar is a weekly meeting in which topics of interest in the theory of computation - broadly construed - are presented. This is sometimes new research by visitors or local people. It is sometimes work in progress, and it is sometimes recent material of others that some of us present in order to learn and share. This is a one-credit seminar which may be taken repeatedly for credit up to six times. 1 credit.

COMPSCI 899 PhD Dissertation

INSTRUCTOR(S):

COMPSCI H311 Honors Colloquium for Introduction to Algorithms

INSTRUCTOR(S): Marius Minea

The design and analysis of efficient algorithms for important computational problems. Emphasis on the relationships between algorithms and data structures and on measures of algorithmic efficiency. Advanced graph algorithms, dynamic programming applications, NP-completeness and space complexity, approximation and randomized algorithms. Experimental analysis of algorithms also emphasized. Use of computer required. Prerequisites: Students must be enrolled in or have completed COMPSCI 311. 1 credit.

COMPSCI H320 Honors Colloquium for Software Engineering

INSTRUCTOR(S): David Fisher

The purpose of this course is to provide students with supplementary material and insights about the software development enterprise. Students meet once a week for a one-hour discussion of software engineering topics whose exploration is intended to provide depth and perspective on the regular material of COMPSCI 320. Topics may be suggested by current events or by problems that may arise in the course of the 320 semester. Students will be required to write a term paper as part of the requirements for this course. Prerequisites: Students must be enrolled in or have completed COMPSCI 320. 1 credit.

COMPSCI H345 Honors Colloquium for Practice and Applications of Data Management

INSTRUCTOR(S): Jaime Davila

Students will study in depth data management topics covered in class. Activities include discussions with the professor and TA, readings, and a project. The project is a semester-long group activity that combines several advanced topics into a practical application. 1 credit.

COMPSCI H377 Honors Colloquium for Operating Systems

INSTRUCTOR(S): Mark Corner

Students taking the honors section will be required to do additional readings and a semester-long project. 1 credit.

INFO 101 Introduction to Informatics

INSTRUCTOR(S): Cheryl Swanier

An introduction to the main concepts of Informatics. There are several 'Big Ideas' in computing, including but not limited to abstraction, data and information, algorithms, programming, the internet, and the global impacts of computing. This class provides an introduction to those ideas and considers some of the ways that those computing principles might be used to solve real world problems. Computer-based assignments are an integral part of this course but no programming knowledge or prior programming experience is expected or required. Not for CS majors. 3 credits.

INFO 150 A Mathematical Foundation for Informatics

INSTRUCTOR(S): Ghazaleh Parvini

Mathematical techniques useful in the study of computing and information processing. The mathematical method of definition and proof. Sets, functions, and relations. Combinatorics, probability and probabilistic reasoning. Graphs and trees as models of data and of computational processes. Prerequisite: R1 math skills recommended. Not intended for Computer Science majors – students interested in a majors-level treatment of this material should see COMPSCI 240 and 250 (or MATH 455). 3 credits.

INFO 190S Introduction to Programming for Information Science

INSTRUCTOR(S): Timothy Richards

The world is rich in data and our ability to process, manipulate, analyze, and visualize that data is growing in power. This course discusses how to understand and solve real-world problems using modern programming techniques. Concepts and techniques covered include the representation of data, fundamentals of programming including data types, expressions, objects, methods, program design, program testing and debugging, state representation, accessing, processing, and manipulating data, generating visualizations of data, interpreting results, and introductory data analytics using data to make predictions that help solve important problems in the world around us. Use of a laptop computer is required. GenEd R2. Prerequisite: R1 (or a score of 20 or higher on the math placement test Part A), or one of the following courses: MATH 101&102 or MATH 104 or MATH 127 or MATH 128 or MATH 131 or MATH 132. 4 credits.

INFO 203 A Networked World

INSTRUCTOR(S): Mohammadhassan Hajiesmaili

The course will cover the technical foundations of today's communication networks, particularly the Internet. It will also address key social, policy, economic and legal aspects of these networks, their use (and abuse), and their regulation. This course covers computer science topics, but all material will be presented in a way that is accessible to an educated audience with or without a strong technical background. Not intended for Computer Science majors – students interested in a majors-level treatment of this material should see COMPSCI 453. 3 credits.