CICS 109 Intro to Data Analysis in R

INSTRUCTOR(S): Jasper McChesney

An introduction to data analysis in the open-source R language, with an emphasis on practical data work. Topics will include data wrangling, summary statistics, modeling, and visualization. Will also cover fundamental programming concepts including data types, functions, flow of control, and good programming practices. Intended for a broad range of students outside of computer science. Some familiarity with statistics is expected. 1 credit.

CICS 110 Foundations of Programming

INSTRUCTOR(S): Kirat Arora, Evan Ciccarelli, Dung Viet Pham, Cole Reilly, Bryn Reimer, Ella Tuson

An introduction to computer programming and problem solving using computers. This course teaches you how real-world problems can be solved computationally using programming constructs and data abstractions of a modern programming language. Concepts and techniques covered include variables, expressions, data types, objects, branching, iteration, functions, classes, and methods. We will also cover how to translate problems into a sequence of instructions, investigate the fundamental operation of a computational system and trace program execution and memory, and learn how to test and debug programs. No previous programming experience required. (Gen. Ed. R2) Prerequisite: R1 (or a score of 15 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.

CICS 127 Intro to Public Interest Tech

INSTRUCTOR(S): Emily Nutwell

Today's world is complex and tech driven. How do we use the tools of information technology to solve problems in a socially responsible way, i.e., in a way that both empowers us and promotes the well-being of the communities in which we live? In this course, we describe the socio-technical world and pragmatic strategies for promoting personal and social responsibility. We explore the questions: What is the public interest in a socio-technical world? What strategies can we use to promote social responsibility in the public sector, private sector, and general public? What can each of us do to make the world a better place? This course is for everyone at all levels and with all interests. No programming or prerequisites are required. We focus on building skills to think analytically, broadly, and strategically, as well as to communicate effectively about complex problems with societal impact. Assignments will provide students multiple paths to success. Counts towards the IT minor and is required for the PIT Undergraduate Certificate. (Gen. Ed. SI) 4 credits.

CICS 160 Object-Oriented Programming

INSTRUCTOR(S): Neha Kennard, Cole Reilly, Ella Tuson

This course will expose students to programming practices beyond the introductory level, concentrating on Object Oriented Programming techniques and an introduction to Data Structures. Students will also study and analyze the complexity of both the algorithms presented in class and of the algorithms they develop. This course also provides experience with the development and analysis of recursive algorithms and programs. Before taking this course, students are expected to have been exposed to the following concepts through a college-level course or equivalent in some high level computer programming language: input and output operations, conditional statements, loops, arrays, recursion, and functions/methods. The course places an emphasis on the careful design and testing of programs. (Gen. Ed. R2) Prerequisite: CICS 110 (previously INFO 190S) or COMPSCI 121 with a grade of C or better. 4 credits.

CICS 210 Data Structures

INSTRUCTOR(S): Mordecai Golin, Marc Liberatore

An introduction to the design, analysis, and implementation of data structures. This course teaches you how to build, test, debug, document, and evaluate objects that encapsulate data and their associated operations using programming constructs and data abstractions of a modern programming language. Concepts and techniques covered include linear and non-linear structures, recursive structures and algorithms, traversal algorithms, binary search trees, balanced trees, priority queues, union-find, hash tables, bloom filters, and graphs. We will also informally compare and contrast the run time efficiency of algorithms and their performance characteristics including the concept of worst-case running time analysis and the classification of algorithms in terms of constant, logarithmic, linear, log linear, quadratic, and exponential time using Big-O notation. (Gen. Ed. R2) Prerequisite: CICS 160 (previously INFO 190T) with a grade of C or better. 4 credits.

CICS 256 Make: Physical Computing

INSTRUCTOR(S): Donghyun Kim

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: CICS 210 (or COMPSCI 187) with a grade of C or better and completion of the R1 (Basic Math Skills) Gen. Ed. 4 credits.

CICS 291C S-Find Strengths/Design Career

INSTRUCTOR(S): Casey Maloney

This course is designed to prepare CICS students for their internship and job searches, improve their professional skills (both technical and soft) and help them approach professional development and/or advanced educational opportunities with confidence. 1 credit.

CICS 291T S-CICS Transfer Success

INSTRUCTOR(S): Casey Maloney

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 305 Social Issues in Computing

INSTRUCTOR(S): Erin Butler, Siobhan Meï, Justin Obara, Christina Sutcliffe

This class satisfies the Junior Year Writing requirement by providing instruction in several different technical communication genres, including research writing and science writing. Writing contexts will include writing for municipal, public, and professional audiences. Through our writing, we will engage with computing topics and challenges facing our current global context and explore various impacts of computers on modern society. Students will produce approximately 20-25 pages of polished written work over the course of the semester. CICS Primary Majors only. Prerequisite: CS Majors: ENGLWRIT 112 with a grade of C or better (or English Writing waiver), COMPSCI 220, COMPSCI 230 and COMPSCI 240 (or 250); INFORM Majors: ENGLWRIT 112 with a grade of C or better (or English Writing waiver) and INFO 248. 3 credits.

CICS 590P TechPolicy & Innov/Common Good

INSTRUCTOR(S): Emily Nutwell, Carolina Rossini

In this course, we will introduce Public Interest Technology, innovation, and relevant regulatory frameworks to provide a foundation to explore the thoughtful design of technology that can serve society while remaining safe for all users and stakeholders. The course begins with examining the core values of public interest technology, such as fairness, transparency, accountability, and inclusivity. We will then review foundational concepts around innovation and strategies, along with an overview of technology governance. An overview of product development models, focusing on Agile product development will be presented along with a presentation of tech governance within an Agile framework. We will then analyze case studies where technologies have either succeeded or failed to uphold these values, examining the societal impact of emerging technologies like artificial intelligence (AI), self-driving cars, digital platforms, and data-driven systems. Throughout the semester, guest lectures from leaders in the fields of technology policy, innovation, and governance will offer insights into how you can bridge the gap between regulatory goals and technological creativity in your future careers. This course counts as an Elective for the CS and INFORM Majors. Cross-listed with SPP 590P. Undergraduate Prerequisite: CICS 305 with a grade of C or better. 3 credits.

COMPSCI 119 Intro to Programming

INSTRUCTOR(S): Aline Weber

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 198C P-Intro/C Programming Language

INSTRUCTOR(S): Meng-Chieh Chiu, 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. 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). A required prerequisite for COMPSCI 230, effective Fall 2023. Prerequisite: CICS 160 (previously INFO 190T or COMPSCI 186 or CICS 210) with a grade of C or better, or COMPSCI 121 with a grade of B or better. 1 credit.

COMPSCI 220 Programming Methodology

INSTRUCTOR(S): Marius Minea, James Perretta

Development of individual skills necessary for designing, implementing, testing and modifying larger programs, including: design strategies and patterns, using functional and object-oriented approaches, testing and program verification, code refactoring, interfacing with libraries. There will be significant programming and mid-term and final examinations. Prerequisite: CICS 210 (or COMPSCI 187) with a grade of C or better. 4 credits.

COMPSCI 230 Computer Systems Principles

INSTRUCTOR(S): Phuthipong Bovornkeeratiroj, 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 data representation, and moves up the stack from there to the features of modern architectures, assembly languages, and operating system services such as I/O, process, and synchronization. This class assumes students have either taken COMPSCI 198C or have equivalent experience in the C programming language. Prerequisite: CICS 210 (or COMPSCI 187) with a grade of C or better and COMPSCI 198C. 4 credits.

COMPSCI 240 Reasoning Under Uncertainty

INSTRUCTOR(S): Justin Domke, Ghazaleh Parvini

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 numbers, and central limit theorem. Probabilistic reasoning -- conditional probability and odds, Bayes' Law, Markov Chains, Bayesian Networks. Statistical topics such as estimation of parameters and linear regression, as time permits. Prerequisite: CICS 160 (previously INFO 190T or COMPSCI 187) or CICS 210 and MATH 132, all with a grade of C or better. 4 credits.

COMPSCI 250 Introduction To Computation

INSTRUCTOR(S): David Barrington, Mordecai Golin

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: CICS 160 (previously INFO 190T or COMPSCI 187 or E&C-ENG 241, or CICS 210) and MATH 132, all with a grade of C or better. 4 credits.

COMPSCI 311 Introduction to Algorithms

INSTRUCTOR(S): Ghazaleh Parvini, Cooper Sigrist

This course will introduce you to a variety of techniques to design algorithms, such as divide and conquer, greedy, dynamic programming, and network flow. 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 and recognize problems that currently do not have efficient algorithms (You will learn about NP-Completeness). There is no programming assignment but you should know a programming language to understand the algorithms . Mathematical experience (as provided by COMPSCI 250) is required. This course is required for the CS Major (BS) and counts as a CS Elective for the CS Major (BA). Prerequisite: CICS 210 (OR COMPSCI 187) , and either COMPSCI 250 or MATH 455, all with a grade of C or better. 4 credits

COMPSCI 320 Software Engineering

INSTRUCTOR(S): Jaime Davila, James Perretta

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 and counts as a CS Elective for the CS Major. Prerequisite: COMPSCI 220 with a grade of C or better. 4 credits.

COMPSCI 325 Intro/Human-Comp Interaction

INSTRUCTOR(S): Ella Tuson

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. This course counts as a CS Elective toward the CS Major and as a Required Core for the INFORM Major. Open to juniors and seniors in Computer Science or Informatics. Prerequisite: Prerequisites: COMPSCI 187 (or CICS 210) with a grade of C or better OR INFO 248 and COMPSCI 186 (or 187 or CICS 160;INFO 190T) with a grade of C or better. 3 credits.

COMPSCI 326 Web Programming

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

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 and an Elective for both the CS and INFORM Majors. Note: as the name 'web programming' denotes, programming is a key component of this class. Previous background in JavaScript is strongly recommended. Prerequisite: CS Majors: COMPSCI 220 with a grade of C or better; INFORM Majors: INFO 248 and CICS 210, both with a grade of C or better. 4 credits.

COMPSCI 345 Pract & Appl of Data Managemnt

INSTRUCTOR(S): Gordon Anderson, Jaime Davila, Neha Makhija

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. The emphasis is on relational databases, though non-relational databases are also introduced. Some of the covered topics include the relational data model, data retrieval, application-driven database design, schema refinement, implementation of basic transactions, database security. This course counts as a CS Elective for the CS Major. Students who have completed COMPSCI 445 are not eligible to take this course without instructor permission. Prerequisite: CS MAJORS: CICS 210 (or COMPSCI 187) with a grade of C or better. INFORM MAJORS: INFO 248 and CICS 160 (previously INFO 190T or COMPSCI 186 or COMPSCI 187), all with a grade of C or better. 3 credits.

COMPSCI 348 Principles of Data Science

INSTRUCTOR(S): Matthew Rattigan

Data science uses various concepts, practices, algorithms, and systems to extract knowledge and insights from data. It encompasses techniques from machine learning, statistics, databases, visualization, and several other fields. When properly integrated, these techniques can help human analysts make sense of vast stores of digital information. This course presents the fundamental principles of data science, familiarizes students with the technical details of representative algorithms, and connects these concepts to applications in industry, science, and government, including fraud detection, marketing, scientific discovery, and web mining. The course assumes that students are familiar with basic concepts and algorithms from probability and statistics. This course counts as a CS Elective toward the CS Major. Prerequisites: CICS 210 (or COMPSCI 187), COMPSCI 240, and COMPSCI 250 (or MATH 455). 3 credits.

COMPSCI 360 Intro Computr & Ntwrk Security

INSTRUCTOR(S): Eugene Bagdasarian

This course provides an introduction to the principles and practice of computer, network, and artificial intelligence security. We will focus on both fundamentals and practical information. Two key topics of this course are communication privacy and security on the web, mobile, and AI. Subtopics include ciphers, hashes, key exchange, security services (integrity, availability, confidentiality, etc.), mobile and web security, prompt injections and jailbreaking, and countermeasures. This course counts as a CS Elective for the CS Major. Prerequisite: COMPSCI 230 with a grade of C or better. 3 credits.

COMPSCI 377 Operating Systems

INSTRUCTOR(S): Phuthipong Bovornkeeratiroj

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. Programming projects in C. This course counts as a CS Elective for the CS Major. Prerequisite: COMPSCI 230 with a grade of C or better. 4 credits.

COMPSCI 383 Artificial Intelligence

INSTRUCTOR(S): Matthew Rattigan

This course aims to give students a high level understanding of the prominent AI topics that are being employed in industry today. It will provide an introduction to each topic, an overview of its supporting algorithms, and examples of products powered by the technology. Particular emphasis will be had on Machine Learning and developing hands-on practical skills with this technology. Upon completion of this course, students will obtain a wider scope of understanding about modern AI trends in software technology and develop an intuition for how this software works. To succeed in this course, students will need a fundamental understanding of data structures and programming fundamentals. Graph and tree data structures will be used in particular. Programming assignments in this class will be done using Python. Experience in at least one programming language is required and it's strongly recommended you have some Python experience before starting. A mathematical foundation in statistics and linear algebra is not strictly necessary but will deepen understanding of course material. This course counts as an Elective for CS and INFORM Majors. Prerequisite: CS MAJORS: (CICS 210 or COMPSCI 187) and COMPSCI 240 (or STATISTC 315), both with a grade of C or better; INFORM MAJORS: INFO 348 and STATISTC 315, both with a grade of C or better. 3 credits.

COMPSCI 389 Intro to Machine Learning

INSTRUCTOR(S): Philip Thomas

The course provides an introduction to machine learning algorithms and applications, and is intended for students with no prior experience with machine learning. Students with prior experience in machine learning or who are already passionate about the subject are encouraged to take COMPSCI 589 instead. Machine learning algorithms answer the question: "How can a computer improve its performance based on data and from its own experience?" The course is roughly divided into thirds: supervised learning (learning from labeled data), reinforcement learning (learning via trial and error), and real-world considerations like ethics, safety, and fairness. Specific topics include linear and non-linear regression, (stochastic) gradient descent, neural networks, backpropagation, classification, Markov decision processes, state-value and action-value functions, temporal difference learning, actor-critic algorithms, the reward prediction error hypothesis for dopamine, connectionism for philosophy of mind, and ethics, safety, and fairness considerations when applying machine learning to real-world problems. This course counts as an Elective for the CS and INFORM Majors. Prerequisite: a grade of C or better in CICS 210 and COMPSCI 240 (or STATISTC 315 or STATISTC 240 or PSYCH 240 or OIM 240 or RES-ECON 212 or SOCIOL 212), and MATH 132. 3 credits.

COMPSCI 390B HarnessingDataSci/SocietalGood

INSTRUCTOR(S): Abhidip Bhattacharyya

This is a project-based course in which students will explore using large-scale datasets and data analysis to address real-world societal problems in domains such as sustainability, health, and work with different techniques of data analysis and processing. Students will address problems of societal or industrial relevance. Each semester, the course will offer one or more real-world datasets and a selection of sample problems and students will define a project based on these datasets to address a real-word problem in a group setting. Students will collaborate in groups for their project. Students will explore modern data processing tools and software systems to build data processing pipelines for their chosen project. Throughout the course, students will be expected to present their project ideas, develop project proposals outlining their implementation plans, and conclude with a final presentation and report submission. This course provides students with the opportunity to integrate their analytical and collaborative skills for real-world problem-solving. This course counts as an Elective for the CS and INFORM Majors. Prerequisites: a grade of C or better in CICS 210 and COMPSCI 220 (or 230 or INFO 248), and either COMPSCI 240, STATISTC 240, PSYCH 240, OIM 240, RES-ECON 212, SOCIOL 212, or STATISTC 315

COMPSCI 420 Software Entrepreneurship

INSTRUCTOR(S): Neena Thota

This course is geared towards students interested in developing software that moves from early stage proof-of-concept ideas towards marketable products with societal benefit. The course leverages the expertise of the Entrepreneurs in Residence (EIR) of the Ventures @ CICS initiative at CICS. The course is grounded in Challenge Based Learning (CBL), an active, student-directed instructional framework that was developed by Apple Inc. and educators. This course counts as a Elective for the CS and INFORM Majors. Prerequisite: COMPSCi 320 (or COMPSCI 326) with a grade of C or better. 3 credits.

COMPSCI 426 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 an Elective for the CS and INFORM Majors. Prerequisite: COMPSCI 320 or COMPSCI 326 with a grade of C or better. 3 credits.

COMPSCI 429 Software Engin Proj Management

INSTRUCTOR(S): Jaime Davila, James Perretta

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 429 will be responsible for: supervising and managing the work of teams of COMPSCI 320 students; interfacing with the other COMPSCI 429 students managing other teams in the course; interfacing with the course instructor, course TA, and course customer. COMPSCI 429 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 429 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 429 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 429 are also expected to be arranged by mutual agreement. An additional one hour weekly meeting of all of the students in COMPSCI 429 is required. This course counts as a CS Elective for the CS Major. 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 B or better. 3 credits.

COMPSCI 445 Information Systems

INSTRUCTOR(S): Trek Palmer

This course is an introduction to the efficient management of large-scale data. The course includes principles for representing information as structured data, query languages for analyzing and manipulating structured data, and core systems principles that enable efficient computation on large data sets. Classical relational database topics will be covered (data modeling, SQL, query optimization, concurrency control), as well as semi-structured data (XML, JSON), and distributed data processing paradigms (e.g. MapReduce and Spark). Additional application topics may include web application development, data integration, processing data streams, database security and privacy. This course counts as an Elective toward the CS Major. Prerequisite: COMPSCI 220 (or 230) and COMPSCI 311 and COMPSCI 345 with a grade of C or better. 3 credits.

COMPSCI 446 Search Engines

INSTRUCTOR(S): Razieh Rahimi

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 Python (other closely related languages may be acceptable). This course counts as a CS Elective for the CS Major. Prerequisite: COMPSCI 240 or COMPSCI 383 with a grade of C or better. 3 credits.

COMPSCI 453 Computer Networks

INSTRUCTOR(S): Arun Venkataramani

This course provides an introduction to fundamental concepts in the design and implementation of computer networks, their protocols, and applications with a particular emphasis on the Internet's TCP/IP protocol suite. Topics to be covered include: overview of network architectures, applications, network programming interfaces (e.g., sockets), transport, congestion, routing, and data link protocols, addressing, local area networks, wireless networks, network security, and network management. There will be five or six homeworks, two programming projects, several hands-on labs (that require an Internet-connected personal computer) and two exams. This course counts as a CS Elective for the CS Major. Prerequisite: Experience programming; COMPSCI 230 (or COMPSCI 377) with a grade of C or better. 3 credits.

COMPSCI 466 Applied Cryptography

INSTRUCTOR(S): Adam O'Neill

This is an undergraduate-level introduction to cryptography. It is a theory course with a significant mathematical component. However, our viewpoint will be "theory applied to practice" in that we will aim to treat topics in a way of applied value. We will discuss cryptographic algorithms used in practice and how to reason about their security. More fundamentally, we will try to understand what security "is" in a rigorous way that allows us to follow sound principles and uncover design weaknesses. The primary topics are: blockciphers, pseudorandom functions, symmetric-key encryption schemes, hash functions, message authentication codes, public-key encryption schemes, digital signature schemes, and public-key infrastructures. This course counts as a CS Elective toward the CS Major. Prerequisite: COMPSCI 311 with a grade of C or better. 3 credits.

COMPSCI 485 Applications of NLP

INSTRUCTOR(S): Katrin Erk

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. Prerequisite: COMPSCI 220 and COMPSCI 240 with a grade of C or better, or LINGUIST 429B (previously LINGUIST 492B) with a grade of C or better. 3 credits.

COMPSCI 491G S-Computer Networking Lab

INSTRUCTOR(S): Parviz Kermani

In this course, students will learn how to put "principles into practice," in a hands-on-networking lab course. The course will cover router, switches and end-system labs in the areas of Single Segment IP Networks, Multiple Segment IP Networks and Static Routing, Dynamic Routing Protocols (RIP, OSPF and BGP), LAN switching, Transport Layer Protocols: UDP and TCP, NAT, DHCP, DNS, and SNMP. Students will also get engaged in evaluating power consumption of network components as an aid in the design of energy efficient (green) networks. This course counts as a CS Elective for the CS Major. Prerequisite: COMPSCI 453 with a grade of C or better. 3 credits.

COMPSCI 501 Formal Language Theory

INSTRUCTOR(S): David Barrington

Introduction to formal language theory. Topics include finite state languages, context-free languages, the relationship between language classes and formal machine models, the Turing Machine model of computation, theories of computability, resource-bounded models, and NP-completeness. This course counts as a CS Elective for the CS Major. Undergraduate Prerequisite: COMPSCI 311 with a grade of C or better. It is recommended that students have a B- or better in 311 in order to attempt 501. 3 credits.

COMPSCI 508 Ethical Consideratns/Computing

INSTRUCTOR(S): Michelle Trim

This course considers an array of ethical issues in computing. Readings and class discussions will cover ethical questions related to artificial intelligence, privacy online, identity in computing, electronic surveillance, environmental responsibility, internet censorship, network policy, algorithmic systems, information access, ownership of intellectual property in computing contexts and others. All examples will be drawn from current and recent events with readings from a range of sources both journalistic and academic. Course assignments will have real world applications and offer students opportunities for developing their speaking and writing skills. Class discussions will be a vibrant component of the course. This course counts as an Elective for the CS and INFORM Majors. 3 credits.

COMPSCI 513 Logic in Computer Science

INSTRUCTOR(S): Marius Minea

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. This course counts as a CS Elective for the CS Major. Undergraduate Prerequisite: COMPSCI 250 and COMPSCI 311. 3 credits.

COMPSCI 514 Algorithms for Data Science

INSTRUCTOR(S): 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. This course counts as a CS Elective for the CS Major. Undergraduate Prerequisite: COMPSCI 240 (or STATISTCS 315/515) and COMPSCI 311 both with a grade of B+ or better, or (COMPSCI 240 and STATISTCS 315/515 and COMPSCI 311 and MATH 233 and MATH 235, all with a C or better). 3 credits

COMPSCI 520 Thry & Practice/Software Engin

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: 1. AI for software engineering including topics like automated testing, program repair and software verification; 2. Software engineering for AI like detecting and fixing fairness bugs in AI models; 3. How industry works. This course counts as a CS Elective for the CS Major. Undergraduate Prerequisite: COMPSCI 320 (or COMPSCI 220 and COMPSCI 326) with a grade of C or better. 3 credits.

COMPSCI 528 Mobile and Ubiquit. Computing

INSTRUCTOR(S): Phuc Nguyen

This course will introduce students to the field of mobile sensing and ubiquitous computing (Ubicomp) – an emerging CS research area that aims to design and develop disruptive technologies with hardware and software systems for real-world messy, noisy and mobile scenarios. The students will learn how to build mobile sensing systems, how to implement it with ubiquitous computing tools, how to make sense of the sensor data and model the target variables. Lastly, the students will learn how to critically think about problems in many application areas including Human-Computer Interaction, Medicine, Sustainability, Transportation, Psychology and Economics, and subsequently practice to find appropriate Ubicomp solutions. There is no exam in this course. The student is expected to work on different hands-on assignments, critique writing, and a final project. This course counts as an Elective toward the CS Major. Open to senior and junior Computer Science majors only Undergraduate Prerequisites: COMPSCI 230 and 240 with a grade of C or better. 3 credits.

COMPSCI 532 Systems for Data Science

INSTRUCTOR(S): Peter Klemperer

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. This course counts as a CS Elective for the CS Major. Undergraduate Prerequisite: COMPSCI 377 and COMPSCI 445 both with a grade of C or better. 3 credits.

COMPSCI 535 Computer Architecture

INSTRUCTOR(S): Charles Weems

The structure of digital computers is studied at several levels, from the basic logic level, to the component level, to the system level. Topics include: the design of basic components such as arithmetic units and registers from logic gates; the organization of basic subsystems such as the memory and I/O subsystems; the interplay between hardware and software in a computer system; the von Neumann architecture and its performance enhancements such as cache memory, instruction and data pipelines, coprocessors, and parallelism. Semester team project to design an architecture and develop a software simulation of it. This course counts as a CS Elective for the CS Major. Undergraduate Prerequisite: COMPSCI 335 with a grade of C or better. 3 credits.

COMPSCI 546 Applied Information Retrieval

INSTRUCTOR(S): Hamed Zamani

COMPSCI 546 is a graduate level course intended to cover information retrieval and other information processing activities, from an applied perspective. There will be numerous programming projects and assignments. It provides a richer technical follow on to COMPSCI 446 (Search Engines) for undergraduates interested in a deeper understanding of the technologies. It also provides a strong basis for continuing on with COMPSCI 646 (Information Retrieval) for those graduate students who are interested in a more complete theoretical coverage of the area. Topics will include: search engine construction (document acquisition, processing, indexing, and querying); learning to rank; information retrieval system performance evaluation; classification and clustering; other machine learning information processing tasks (e.g. basic deep learning models for information retrieval); and many more. This course counts as a CS Elective for the CS Major. Undergraduate Prerequisite: COMPSCI 320 (or COMPSCI 326) and COMPSCI 383 (or COMPSCI 389 or COMPSCI 446 or COMPSCI 485 or COMPSCI 585) all with a grade of C or better. 3 credits.

COMPSCI 561 System Defense and Test

INSTRUCTOR(S): Parviz Kermani

This class trains students to detect and analyze weaknesses and vulnerabilities in target systems as a method of assessing the security of a system. We focus on tools and techniques that an attacker would employ but from the perspective of an ethical system administrator. Topics include tools and techniques for penetration testing and attacks, information gathering, social engineering, and defenses. Specific topics include malware, denial of service attacks, SQL injection, buffer overflow, session hijacking, and system hacking, network sniffing and scans, wireless encryption weaknesses and other WiFi issues, IDS/firewall evasion, metasploit tools, physical security, and setting up honeypots. Previously INFOSEC 690S. This course counts as a CS Elective for the CS Major. Undergraduate Prerequisite: COMPSCI 360 (or COMPSCI 460 or COMPSCI 560 or COMPSCI 660 or E&C-ENG 371) and COMPSCI 453 (or E&C-ENG 325), all with a grade of C or better. 3 credits.

COMPSCI 564 Cyber Effects

INSTRUCTOR(S): Nick Merlino, Edward Walters

This course covers a broad range of topics related to cyber security and operations. Our focus is on real world studies of reverse engineering, exploit analysis, and capability development within the context of computer network operations and attack. The course has an emphasis on hands-on exercises and projects. Topics covered include computer architecture and assembly language, principles of embedded security, the essentials of exploit development and analysis (including using industry standard tools such as Ghidra, and utilizing computer security databases such as CVE), and discussion of real-world events and techniques. This course counts as a CS Elective for the CS Major. Undergraduate Prerequisite: COMPSCI 230 (or E&C-ENG 322) and COMPSCI 360 (previously COMPSCI 460 or COMPSCI 365 or COMPSCI 390R or COMPSCI 466 or E&C-ENG 371), all with a grade of C or better. 3 credits.

COMPSCI 589 Machine Learning

INSTRUCTOR(S): Bruno Castro da Silva

This course will introduce core machine learning models and algorithms for classification, regression, clustering, and dimensionality reduction. On the theory side, the course will focus on effectively using machine learning methods to solve real-world problems with an emphasis on model selection, regularization, and empirical evaluation. The assignments will involve both mathematical problems and implementation tasks. The course will assume experience with the Python programming language. While this course has an applied focus, it still requires strong mathematical background in probability and statistics, calculus, and linear algebra. Graduate students should check the descriptions of the undergraduate prerequisite courses to verify that they have sufficient mathematical background for 589. Graduate students from outside computer science with sufficient background are also welcome to take the course. This course counts as a CS Elective for the CS Major. Undergraduate Prerequisite: MATH 545 and COMPSCI 240 and STATISTC 315/515 all with a grade of C or better. (MATH 545 can be skipped by students who have taken MATH 235 and MATH 233 both with B+ or better. STATISTC 315/515 can be skipped by students who have taken COMPSCI 240 with a B+ or better). 3 credits.

COMPSCI 590AB Quantum Cryptography

INSTRUCTOR(S): Filip Rozpedek

The ability to transmit quantum information over long distances will enable implementation of many fascinating quantum communication tasks and provide us with novel capabilities that reach beyond what we can do over classical Internet alone. Examples of such tasks include blind quantum computing, clock synchronization or distributed quantum computing. Quantum cryptography is one family of such tasks with the most famous one being quantum key distribution. This task, which is currently the most mature quantum technology, enables distribution of shared keys through a protocol that is information-theoretically secure and whose security remarkably is guaranteed by the laws of quantum physics. Such unconditional security cannot be achieved in the classical world. In this course we will introduce the world of quantum cryptographic protocols and describe how the power of quantum mechanics can enable distribution of shared secret keys even with untrusted devices. It will also introduce many other fascinating quantum protocols beyond quantum key distribution. The course will meet jointly with the course 690BB which covers the same material. Unlike 690BB this course does not include a project as one of the assessment components and it puts more grade weight on the homework. Students who take this course cannot simultaneously, nor later, enroll in 690BB. This course counts as a CS Elective for the CS Major. Undergraduate Prerequisite: COMPSCI 412/490Q (OR COMPSCI 590Q OR COMPSCI 590QC OR E&C-ENG 550 OR PHYSICS 537) with a grade of C or better. 3 credits

COMPSCI 590AE Mobile and Wireless Networks

INSTRUCTOR(S): James Kurose

This course covers wireless networking and mobility principles and practice. The focus of "practice" is primarily 802.11 (WiFi) and 4G/5G cellular networks. Unlike many other wireless networking courses, this course focuses primarily on the networking and systems aspects of wireless and mobile networks, rather than on the wireless channel aspects. This course counts as a CS Elective for the CS Major. Undergraduate Prerequisite: COMPSCI 453 or E&C-ENG 374 with a grade of C or better. 3 credits.

COMPSCI 590QC Quantum Communication

INSTRUCTOR(S): Gayane Vardoyan

This course will provide an introduction to fundamental principles underlying quantum communication, including quantum repeaters, entanglement, and other tools needed for the implementation of distributed quantum protocols. Topics include: basics of quantum computation and information; modeling and coping with noise in quantum systems; entanglement generation schemes; quantum Internet and its applications. This course counts as a CS Elective for the CS Major. Undergraduate Prerequisite: MATH 235 and COMPSCI 240 (or STATISTC 315/515), both with a grade of C or better. 3 credits

COMPSCI 590RM

INSTRUCTOR(S): David Jensen

This course introduces concepts, practices, and tools for conducting effective research. You will learn how to read technical papers, interpret published research, assess the research frontier, select research topics, devise research questions and hypotheses, propose and plan research activities, analyze experimental results, and report those results. The course is structured around four activities: (1) Synchronous and asynchronous lectures on basic research strategies and techniques; (2) Synchronous activities that apply course concepts; (3) Reading and discussions of technical papers in computer science; and (4) An individual semester-long project that replicates an existing published study in computer science. The course requires significant reading, reviewing, and writing. Students are expected to participate actively in class activities. The course is intended for MS and advanced undergraduate students. It provides a grounding in research methods that will aid your entry into research-oriented industrial positions and PhD studies. For undergraduates considering graduate studies, this course will help inform and accelerate that direction. For undergraduates, 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. Undergraduates must obtain approval of the Computer Science Honors Program Director prior to registering. Does not count as a CS elective for the CS major (BA or BS).   3 credits.

COMPSCI 602 Research Methods/Empirical CS

INSTRUCTOR(S): David Jensen

This course introduces concepts, practices, and tools for conducting effective research. You will learn how to read technical papers, interpret published research, assess the research frontier, select research topics, devise research questions and hypotheses, propose and plan research activities, analyze experimental results, and report those results. The course is structured around five activities: (1) Synchronous and asynchronous lectures on basic research strategies and techniques; (2) Synchronous activities that apply course concepts; (3) Reading and discussions of technical papers in computer science; (4) An individual semester-long empirical research project; and (5) Review and feedback on other student's projects. The course requires significant reading, reviewing, and writing. Students are expected to participate actively in class activities and to provide meaningful comments on the work of other students. For PhD students, this course will help accelerate your current and future research. For MS students, this course will provide a grounding in research methods that will aid your entry into research-oriented industrial positions and PhD studies. For undergraduates considering graduate studies, this course will help inform and accelerate that direction. For undergraduates, 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. Undergraduates must obtain approval of the Computer Science Honors Program Director prior to registering. Open to graduate Computer Science students only. 3 credits.

COMPSCI 603 Robotics

INSTRUCTOR(S): Hao Zhang

This course is intended to serve as an advanced overview of robotics spanning the complete autonomy loop: perception, planning, and control. We will study the theory, algorithms, and efficient implementations related to these topics, with focus on open discussions for how to do research to go beyond the state of the art. Students will gain hands-on experience in implementing, and extending such algorithms using simulations. 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. Open to graduate Computer Science students only. Prerequisite: The mathematical maturity expected of incoming Computer Science graduate students, knowledge of algorithms at the level of COMPSCI 311. 3 credits.

COMPSCI 613 Adv Logic in Computer Science

INSTRUCTOR(S): Marius Minea

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 614 RandomizedAlgorithms/App to DS

INSTRUCTOR(S): Andrew McGregor

Randomness has proven itself to be a useful resource for developing provably efficient algorithms and protocols for large scale data processing. As a result, the study of randomized algorithms has become a major research topic in recent years. This course will explore a collection of techniques for effectively using randomization and for analyzing randomized algorithms, as well as examples from a variety of settings and problem areas. The course is a natural follow on both COMPSCI 514: Algorithms for Data Science and COMPSCI 611: Advanced Algorithms. 3 credits.

COMPSCI 620 AdvSoftwareEng:Synth&Develpmnt

INSTRUCTOR(S): Madeline Endres

This course examines the varied approaches and human factors that influence software development. We investigate various ideas about how software products can be structured and function. Topics include, but are not limited to, software development productivity, requirements elicitation, software development process, program comprehension, software development team structure, and emerging software development paradigms. Students will engage with both established best practices and the latest research through a series of readings, in-class presentations, and a final project as part of their coursework. 3 credits.

COMPSCI 625 Advanced Methods in HCI

INSTRUCTOR(S): Narges Mahyar

This is an advanced course in HCI. This course will provide a deeper treatment of some topics that are typically found in an undergraduate HCI course. For example, design methodologies, evaluation methodologies (both quantitative and qualitative), human information processing, cognition, and perception. This course will also introduce students to research frontiers in HCI. The course will cover topics of Universal Usability, CSCW, Digital Civics and fundamentals of designing interactive technology for people. 3 credits.

COMPSCI 627 Fixing Social Media

INSTRUCTOR(S): Ethan Zuckerman

Over the past decade, user-generated participatory media – social media – has emerged as the dominant model for content of the Internet. From Facebook to Twitter, YouTube to Wikipedia, content created by non-professionals and circulated for commercial and non-commercial motives underpins seven of the top 10 websites in the US, and has become an increasingly important component of the news ecosystem. While social media was initially hailed as a powerful tool for broadening civic participation, many problems have emerged with the rise of the medium, from questions of whether social media usage is bad for our individual mental health, to whether the fabric of our democracy is being damaged by disinformation, fragmentation and hyperpolarization. As legislators look to regulate these platforms and commentators propose shutting them down entirely, this course looks for an alternative: affirmative visions of social media that are good for individuals and society, which we could work towards building. This class examines possible problems with existing modes of social media, discusses ways in which social media could be a benefit to individuals and societies, develops case studies of successful and healthy online communities, and ultimately designs and builds tools to improve existing social media systems or replace them with novel models. Students will write reflectively about weekly readings and discussions and participate in multi-week projects, ultimately building teams to work on final projects. Meets with COMM 627 and SPP 627. 3 credits.

COMPSCI 645 DatabaseDesign&Implementation

INSTRUCTOR(S): Marco Serafini

This course covers the design and implementation of traditional relational database systems as well as advanced data management systems. The course will treat fundamental principles of databases such as the relational model, conceptual design, and schema refinement. We will also cover core database implementation issues including storage and indexing, query processing and optimization, and transaction management. Additionally, we will address challenges in modern information systems, including scalability and parallelization, big data systems, cloud databases, dealing with graph and vector data, and the integration of database management systems and machine learning. 3 credits.

COMPSCI 651 Optimization/Computer Science

INSTRUCTOR(S): Madalina Fiterau Brostean

Much recent work in computer science in a variety of areas, from game theory to machine learning and sensor networks, exploits sophisticated methods of optimization. This course is intended to give students an in-depth background in both the foundations as well as some recent trends in the theory and practice of optimization for computer science. The Optimization course covers these topics, which are critical to a large number of research projects conducted within the department. 3 credits.

COMPSCI 677 Distributed&Operating Systems

INSTRUCTOR(S): Peter Klemperer

This course provides an in-depth examination of the principles of distributed systems and advanced concepts in operating systems. Covered topics include client-server programming, distributed scheduling, virtualization, cloud computing, distributed storage, security in distributed systems, distributed middleware, ubiquitous computing, and applications such as the Internet of Things, Web and peer-to-peer systems. Prerequisites: Students should be able to easily program in a high-level language such as Java, C++ or Python, have had a course on data structures, be familiar with elements of computer architecture and have had previous exposure to the operating system concepts of processes, virtual memory, and scheduling. A previous course on uniprocessor operating systems (e.g., COMPSCI 377) will be helpful but not required. 3 credits.

COMPSCI 682 Neural Networks: Modern Intro

INSTRUCTOR(S): Subhransu Maji

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 Pytorch. Required background includes Linear Algebra, Probability and Statistics, and Multivariate Calculus. All assignments will be in the Python programming language. Open to graduate Computer Science students only. 3 credits.

COMPSCI 683 Artificial Intelligence

INSTRUCTOR(S): Yair Zick

In-depth introduction to Artificial Intelligence focusing on techniques that allow intelligent systems to reason effectively with uncertain information and cope limited computational resources. Topics include: problem-solving using search, heuristic search techniques, constraint satisfaction, local search, abstraction and hierarchical search, resource-bounded search techniques, principles of knowledge representation and reasoning, logical inference, reasoning under uncertainty, belief networks, decision theoretic reasoning, representing and reasoning about preferences, planning under uncertainty using Markov decision processes, multi-agent systems, and computational models of bounded rationality. 3 credits.

COMPSCI 688 Probabilistic Graphical Models

INSTRUCTOR(S): Benjamin Marlin

Probabilistic graphical models provide an intuitive language for describing the structure of joint probability distributions using graphs. They enable the compact representation and manipulation of exponentially large probability distributions, which allows them to efficiently manage the uncertainty and partial observability common in real-world problems. As a result, graphical models have become invaluable tools in a wide range of areas from computer vision and sensor networks to natural language processing and computational biology. The aim of this course is to develop the knowledge and skills necessary to effectively design, implement and apply these models to solve real problems. The course will cover (a) Bayesian and Markov networks; (b) exact and approximate inference methods; (c) estimation of the parameters and structure of graphical models; (d) broader topics in probabilistic inference for statistics and machine learning. Students entering the class should have good programming skills and knowledge of algorithms. Undergraduate-level knowledge of probability and statistics and a prior machine learning course are recommended. Open to graduate Computer Science students only. 3 credits.

COMPSCI 690AB Systems for Deep Learning

INSTRUCTOR(S): Shiqing Ma

This course is designed to provide a comprehensive understanding of computer systems architecture that supports deep learning workloads. It assumes students have prior knowledge on computer systems, algorithms, and Python/C/C++ programming background. In the course, we will study the full-stack system design to support deep learning, covering topics from the high-level programming frameworks to low-level kernel implementations. We will also introduce cutting-edge research on efficient and scalable deep learning model training, inference, and serving. 3 credits.

COMPSCI 690AG Algorithmic Game Theory

INSTRUCTOR(S): Hedyeh Beyhaghi

This course offers a graduate-level introduction to topics at the interface of theoretical computer science and economics, with a focus on strategic and learning-based interactions. We will cover the basics of mechanism design, auctions, and pricing, as well as models of strategic behavior such as Nash equilibrium and the price of anarchy. Additional topics include learning in games, strategic classification, and dynamics in multi-agent systems. While COMPSCI 515 focuses on cooperation, social choice, and fair division, this course emphasizes competitive settings and mechanism design. Students interested in both perspectives are encouraged to take both courses, which are designed to be complementary; however, prior completion of COMPSCI 515 is not required. 3 credits.

COMPSCI 690BB Quantum Cryptography

INSTRUCTOR(S): Filip Rozpedek

The ability to transmit quantum information over long distances will enable implementation of many fascinating quantum communication tasks and provide us with novel capabilities that reach beyond what we can do over classical Internet alone. Examples of such tasks include blind quantum computing, clock synchronization or distributed quantum computing. Quantum cryptography is one family of such tasks with the most famous one being quantum key distribution. This task, which is currently the most mature quantum technology, enables distribution of shared keys through a protocol that is information-theoretically secure and whose security remarkably is guaranteed by the laws of quantum physics. Such unconditional security cannot be achieved in the classical world. In this course we will introduce the world of quantum cryptographic protocols and describe how the power of quantum mechanics can enable distribution of shared secret keys even with untrusted devices. It will also introduce many other fascinating quantum protocols beyond quantum key distribution. The course will meet jointly with the course 590AB which covers the same material. Unlike 590AB this course includes a project as one of the assessment components and it puts less grade weight on the homework. Students who take this course cannot simultaneously, nor later, enroll in 590AB. This course requires previous knowledge of the basics of quantum information. Students enrolling in this course need to be familiar with the following concepts covered in COMPSCI 412 (Quantum Information Science) and COMPSCI 590Q/648 (Quantum Information Systems) and corresponding mathematical formalism: 1) Pure states of quantum systems and qubits in particular 2) Projective measurements in a basis 3) Unitary transformations 4) Bloch sphere representation of a qubit 5) Basics of the density matrix formalism. 3 credits.

COMPSCI 690G Security/Large-Scale Systems

INSTRUCTOR(S): Pubali Datta

This course provides an in-depth examination of the issues in system security, and assumes prior knowledge of operating system concepts. We will start with learning the classic approaches to security attacks and defenses from the perspective of a single host system. Topics include access control, information flow control, system auditing, privilege separation, security policies, host-based intrusion detection etc. Then we will study how these classic mechanisms are expanded and adapted to modern distributed system platforms such as cloud platforms, Internet of Things platforms, and large enterprises. 3 credits.

COMPSCI 690QC Quantum Communication

INSTRUCTOR(S): Gayane Vardoyan

This course will provide an introduction to fundamental principles underlying quantum communication, including quantum repeaters, entanglement, and other tools needed for the implementation of distributed quantum protocols. Topics include: basics of quantum computation and information; modeling and coping with noise in quantum systems; entanglement generation schemes; quantum Internet and its applications. 3 credits

COMPSCI 690R Computing/DigiBiomarkersinHC

INSTRUCTOR(S): Sunghoon Lee

Computer science has played a pivotal role in developing innovative technologies to monitor patients' health conditions beyond the traditional laboratory or clinical environments. The development of clinically valid digital biomarkers can enable a wide range of applications, including the implementation of targeted behavioral interventions to drive substantial improvements in health-related outcomes, monitoring the progress of individuals undergoing interventions, and assessing the efficacy of emerging therapeutic interventions. In this course, we will delve into the application of machine learning and mobile technologies in the development of digital biomarkers. Focusing on its application in the analysis of human movements, behaviors, and behavioral phenotypes, the course curriculum will encompass the fundamentals of human movement analysis, mobile and wearable sensing technologies to support remote human movement monitoring, fundamentals of signal processing techniques, and state-of-the-art machine learning techniques. Additionally, the course examines the theoretical foundations and practical applications of biomedical measurement sciences, and how they integrate with modern machine learning research to develop reliable, responsive, and clinically valid digital biomarkers for movement-related health and wellness. This course contains lectures, assignments, a final project, paper presentations, and critical discussions. 3 credits.

COMPSCI 690S AI Alignment

INSTRUCTOR(S): Scott Niekum

This course will focus on modern machine learning approaches to learn from human demonstrations, preferences, feedback, and other multimodal signals, with the goal of aligning agent goals and behaviors with human values and desires. For the purposes of both safety and practicality, it is increasingly important for AI systems to be well-aligned with human users as their capabilities improve and they are deployed more frequently in real-world settings. This course will provide the basic tools to address these important issues, covering topics such as behavioral cloning, inverse reinforcement learning, preference elicitation, active learning, learning from feedback, value alignment, bounded rationality, and best practices for human studies. We will examine applications including robotics, large language models, and self-driving cars. 3 credits.

COMPSCI 690U Computational Bio & Bioinform

INSTRUCTOR(S): Anna Green

This course is designed to provide computer scientists with a comprehensive introduction to the field of computational biology. The course will cover the application of computational techniques to modern research challenges in biology, discussing both foundational algorithms and newly introduced methods. The necessary background on biology will be provided in order to contextualize the methods. The primary focus will be analysis of genomic data, including variant calling, genome annotation, sequence search, sequence alignment, phylogeny construction, mutation effect prediction, population genetics, and genotype-phenotype association studies. We will also cover protein structure analysis and prediction. Throughout the course, we will emphasize the unique challenges to working with biological data. Through lectures and hands-on programming problem sets, students will develop the necessary skills to tackle computational challenges in the field of biology. 3 credits.

COMPSCI 691O S-Tools/Explanatory&TutorSystm

INSTRUCTOR(S): Beverly Woolf

Artificial Intelligence will radically change education. Through machine learning, data mining, analytics, robotics, and user models, AI will replace false learning boundaries (e.g., learning places, time, level of study); personalize learning; make learning instantly available to everyone; connect learners with partners; provide multi-media; and augment human learning ability. This seminar examines recent work in explanatory and tutoring systems, presents theories about digital teaching and learning, and describes how to deliver personalized teaching in online systems. Such software supports people working alone or in collaborative inquiry to rapidly access and integrate global information. This course describes how to build tutors, stimulates awareness of research issues, and promotes sound analytic and design skills. Specific topics include systems that support collaboration, inquiry, natural language dialogue, authoring tools and user models. The course is appropriate for students from many disciplines (e.g., computer science, linguistics, education, and psychology), researchers, and practitioners from academia, industry, and government. No programming is required. Students will read and critique papers about AI tools (e.g., vision, natural language), methods, and will study the complexity of human learning through advances in cognitive science. Weekly assignments invite students to critique the literature and a final project requires a detailed specification (not a program) for a tutor on your chosen topic. Students present readings from the research literature and several working systems will be available for hands-on or virtual critique. 3 credits.

COMPSCI 692CT S-Category Theory for AGI

INSTRUCTOR(S): Sridhar Mahadevan

The world's largest AI companies are collectively spending several trillion dollars in the most expensive race in human history to build AGI systems. What is AGI and how close are we to achieving it? This course will introduce students to the use of the latest advances in pure mathematics from the 21st century, such as Markov symmetric monoidal categories in causality, probability and statistics, to analyze the current AGI systems, and hopefully develop the next-generation of more sophisticated AGI systems. It will teach students the core concepts — categories and functors, universal representations and Yoneda Lemma, limits and colimits, adjoint functors, monads and categorical probability, Kan extensions — as well as applications of these abstract concepts in generative AI and AGI systems, machine learning approaches such as clustering and dimensionality reduction, causal reasoning and probabilistic inference, modeling RL over universal coalgebras, relational knowledge representation using categories, and designing safe AI systems using topos theory. The theoretical ideas will be fully co-developed with hands-on experience in programming categorical methods on the current AGI systems, such as GPT5, Claude, Gemini etc. Students will only be required to have access to the free versions of these chatbots. 3 credits.

COMPSCI 692EF S-Embedded Models/IoT & Health

INSTRUCTOR(S): Deepak Ganesan, Phuc Nguyen

The deployment of foundation models in resource-constrained mobile and wearable devices presents unique challenges and opportunities, particularly in health and wearable sensing applications. This advanced seminar explores the cutting-edge intersection of large AI models, embedded systems, and digital health, with a focus on real-world applications in wearable and mobile biosignal sensing platforms. The course examines two primary themes: (1) the application of foundation models in multimodal biosignal sensing contexts, including real-time physiological monitoring, neural signal processing, behavioral understanding, and clinical decision support through wearable devices such as EEG headsets, smartwatches, smart rings, eyewear, and earables; and (2) the technical challenges and solutions for deploying these powerful models on resource-constrained devices, including model compression, efficient inference, on-device learning, and real-time processing requirements. Example topics will include: Application Domains in Health and Wearables, Systems Optimizations and Architecture, IoT Applications and Environments. The course will primarily consist of paper readings, presentations, and discussions. Students will critically examine recent advances in the field through both academic papers and industry developments. For 3-credit option, a semester-long research project is required, involving either the development of a novel application using foundation models on wearable platforms or the implementation of optimization techniques for embedded deployment. 1-3 credits.

COMPSCI 692G S-Simulation & Casual Modeling

INSTRUCTOR(S): Peter Haas, David Jensen

This seminar will explore the relationships between simulation modeling and causal modeling, focusing on the potential of these two research fields to enrich each other. From a simulation point of view, causal models could provide methods for enabling new kinds of tractable probabilistic reasoning about models, for learning models directly from data, and for learning high-level abstractions of a detailed mechanistic model. From a causal-modeling viewpoint, simulation modeling could provide a means to enhance the expressiveness of languages for high-level statistical models with finer-grained mechanistic and generative semantics. Topics will include simulation, causal learning and reasoning, model-based RL, simulation metamodeling, probabilistic programming, process mining, and discovery of causal mechanisms. 1-3 credits.

COMPSCI 692L S-Natural Language Processing

INSTRUCTOR(S): Brendan O'Connor

Weekly seminar requiring students to read an NLP paper and discuss and review it from a variety of perspectives. Some weeks will feature invited speakers instead of paper reviews. 1 credit.

COMPSCI 692QA S-Quantum & Artificial Intel

INSTRUCTOR(S): Justin Domke, Stefan Krastanov

The seminar will focus on rigorous results on complexity and learnability of modeling physical processes, and applications of quantum information science to understanding of AI and vice versa. 1 credit.

COMPSCI 692QB S-Quantum Networking

INSTRUCTOR(S): Stav Haldar, Filip Rozpedek

This seminar will cover topics in quantum networks and will be based on invited talks given by invited speakers supplemented by a few discussion sessions. Possible topics include quantum network architectures, quantum repeater design, techniques for overcoming noise in a network including entanglement distillation and quantum error correction, resource allocation in quantum networks, various quantum network applications as well as performance modeling and analysis of quantum networks. Specific topics covered will depend on the expertise of the invited speakers. The first objective of this seminar is to bring researchers in quantum information up to speed regarding the state of the art in quantum networks. The second objective is to identify MS and PhD level research problems in quantum networks. The seminar will be done in collaboration with the NSF Center for Quantum Networks. May be taken repeated for credit up to three times. 1 credit.

COMPSCI 701 Advanced Topics Computer Sci

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, 701 followed by 701Y.
The 6 credit option is for a project that will be completed over two semesters with enrollment in only one semester.

COMPSCI 701Y Advanced Topics Computer Sci

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 791U S-Adv Topics in Info Retrieval

INSTRUCTOR(S): Razieh Rahimi, Hamed Zamani

A seminar in which students will read, present, and discuss research papers on recent and advanced topics in Information Retrieval (IR). Students are expected to read papers for every session. For one or more sessions in the semester, students are expected to make summary presentations and lead discussion of the papers. Students should have taken COMPSCI 646, Information Retrieval, or a comparable course. This semester, the seminar will primarily cover the following main topics: Retrieval-Enhanced Machine Learning, Large Language Models for IR, Reasoning for Information Retrieval, and Agentic Information Retrieval. Along with studying research papers on the aforementioned topics, students registered for three credit hours should work with the instructor to complete a mutually agreed-upon research project. This seminar assumes prior knowledge of fundamental information retrieval concepts. 1 or 3 credits.

COMPSCI 879 TeachngAssist/TomorrowsFaculty

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 two credit (not repeatable) course is semester long and taken by all TAs prior to assuming assistantship. 2 credits.

COMPSCI 891M S-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 seminar may be taken repeatedly for credit up to six times. 1 credit.

COMPSCI H230 Comp Syst Principles colloq

INSTRUCTOR(S): Meng-Chieh Chiu

Students will explore topics from the 230 curriculum in greater detail, focusing on low-level systems concepts, memory and process management, and concurrency. The course emphasizes deeper understanding of how systems behave at the machine level through additional reading, discussion, and implementation. Students will also gain hands-on experience with an alternative systems programming language (e.g., Rust), enabling them to compare design choices and safety models across system languages. 1 credit.

COMPSCI H311 Intro to Algorithms (colloq)

INSTRUCTOR(S): Ghazaleh Parvini

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. Prerequisite: CICS 210 (or COMPSCI 187) and COMPSCI 250 (or MATH 455) all with a grade of C or better (or COMPSCI 311 with a grade of B or better). 1 credit.

COMPSCI H466 Applied Cryptography colloq

INSTRUCTOR(S): Adam O'Neill

This colloquium will guide students in an in-depth study of both theoretical and practical cryptographic principles. Topics include zero-knowledge proofs, derivatives of which are now used in anonymous cryptocurrency such as ZCash, secure messaging protocols (e.g. Signal, Whatsapp), and secure multiparty computation. Other topics at student request. Prerequisite: COMPSCI 311 with a grade of C or better (or COMPSCI 466 with a grade of B or better). 1 credit.

COMPSCI H589 Machine Learning (colloq)

INSTRUCTOR(S): Bruno Castro da Silva

This colloquium will enrich the primary course by focusing on reading, presenting, and discussing foundational and recent research papers from the machine learning literature. Students will write weekly reading responses, and lead one to two group discussions over the course of the semester. Prerequisite: Students must be enrolled in or have completed COMPSCI 589. 1 credit.

INFO 101 Introduction to Informatics

INSTRUCTOR(S): Michelle Trim

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. Open to undergraduate students NOT majoring in Computer Science. 3 credits.

INFO 150 Mathmtcl Fndtn for Informatics

INSTRUCTOR(S): Peter Haas

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 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 CS majors-level treatment of this material should see COMPSCI 453. 3 credits.

INFO 248 Intro to Data Science

INSTRUCTOR(S): Gordon Anderson

This course is an introduction to the concepts and skills involved with the collection, management, analysis, and presentation of data sets and the data products that result from the work of data scientists. Privacy, algorithmic bias and ethical issues are also discussed. Students will work with data from the financial, epidemiological, educational, and other domains. The course provides examples of real-world data that students work with using various software tools. This course consists of two lecture meetings and one lab meeting per week. Readings will be assigned as preparation for each class meeting. A semester project will be assigned. Students work in pairs to develop their project over the semester. The project provides students with an opportunity to work collaboratively to explore the topics in more depth in a specialized domain. A midterm and final exam will be given. Grades are determined by a combination of scores on lab activities, projects, and exam scores. Software: all software is freely available. Open to INFORM majors. Prerequisite: a grade of C or or above in the following courses: CICS 110 (or CICS 160 or COMPSCI 119 or COMPSCI 121) with a grade of C or above and either: PSYCH 240, OIM 240, STATISTC 240, RES-ECON 212, SOCIOL 212, OR STATISTC 315/515, OR COMPSCI 240, with a grade of C or above. 4 credits.

INFO 348 Data Analytics with Python

INSTRUCTOR(S): Matthew Rattigan

The modern world is awash with data, and making sense of it requires specialized skills. This course will expose students to commonly used data analytics techniques. Topics include the acquisition, manipulation, and transformation of structured data, exploratory data analysis, data visualization, and predictive modeling. Students in this course will learn and use the Python programming language and tools for working with data. Analysis will be performed using real data sets. Does not count as a CS Elective (BA or BS). Satisfies one of the Data Science Concentration requirements and counts as an elective for the Health and Life Sciences Concentration for the Informatics major. Open to INFORM majors. Prerequisite: INFO 248 and CICS 160 (or INFO 190T or COMPSCI 186 or 187), both with a grade of C or better. 3 credits.