College of Information and Computer Sciences
UMassAmherst

Course Descriptions
2017 Spring

CICS 269 Trust Assurance and Cybersecurity Capstone

INSTRUCTOR(S):

CICS 597C Special Topics - Introduction to Computer Security

INSTRUCTOR(S):

This course provides an introduction to the principles and practice of computer and network security with a focus on both fundamentals and practical information. Topics include ethics, primary definitions, applied cryptography, networking (e.g., firewalls, wireless security, routing security), operating system hardening, malware and other attacks, and incident handling. The course will also cover security topics trending in current events and recent research. This course involves discussions of computing concepts and computer configuration. Students are presumed to have already experienced operating computers and a basic familiarity with computing concepts. Students who have taken an equivalent of INFO 190IN (or INFO 150), COMPSCI 187 (or COMPSCI 190D), and COMPSCI/INFO 290NW(203) should be well prepared. Open to Juniors/Seniors. Instructor permission is required for students lacking prerequisites. 3 credits.

CICS 597P Special Topics - Trust Assurance and Cybersecurity Capstone

INSTRUCTOR(S):

COMPSCI 105 Computer Literacy

INSTRUCTOR(S): William Verts

Microcomputers are used widely in all areas of modern life. For this reason it is important for all students to understand how computers work and how computers can be used as a problem-solving tool. The focus of this course is on computer applications. The course stresses the ways in which computers can help you solve problems efficiently and effectively. The course provides a broad introduction to hardware, software, and mathematical aspects of computers. Four application areas are discussed: Internet tools (including Web page design), word processing, spreadsheets, and databases. Weekly lab assignments are an integral part of the course, and it is expected that students have access to their own computing equipment. There are optional lab times set up for students who do not have the proper equipment or software available to them. This course is a "Foundations" course for the Information Technology minor. Students who are more interested in computer programming should take a course such as COMPSCI 119 or COMPSCI 121. Prerequisites: reasonable high school math skills. Typing ability is also an important asset for the course. Some previous computer experience, while not absolutely required, will prove helpful. Not for CS majors. 3 credits.

COMPSCI 119 Introduction to Programming

INSTRUCTOR(S): William Verts

This introduction to computer programming with Python emphasizes multimedia (graphics and sound) applications that are relevant for Web designers, graphic artists, and anyone who just wants to have more fun with their computer. Students will explore basic concepts in computer science and computer programming by manipulating digital images and sound files. No prior programming experience is needed. Not for CS majors. 3 credits.

COMPSCI 121 Introduction to Problem Solving with Computers

INSTRUCTOR(S): Gordon Anderson, Neena Thota

CMPSCI 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, arrays, graphics, and GUIs. No previous programming experience required. A companion introduction to programming class, COMPSCI 119 is also offered. If you are fairly sure you only want to do just one programming class, take that course; if you think it likely that you will do more than one programming course, take 121. Use of computer is required. Prerequisite: R1. 4 credits.

COMPSCI 145 Representing, Storing, and Retrieving Information

INSTRUCTOR(S): William Verts

An introductory course in the use of data in computer systems, a core course for the Information Technology certificate. Formats for representing text, numbers, sound, images, etc., as strings of bits. Equations of lines and curves, modeling of synthetic scenes (i.e., ray tracing), exploring the frequency domain and holography. Basic information theory, use and limitations of file compression and encryption. Structured databases and how to use them. Information retrieval in heterogenous environments such as the Web. XML as a language for defining new formats for representing data. Review of historical, pre-computer methods of information representation. Prerequisites: "Basic computer literacy", i.e., user-level familiarity with a modern operating system and some experience with application programs. Tier I math skills. Recommended for First Year and Sophomore Non-Majors.   Prerequisite: R1. 3 credits.

COMPSCI 186 Using Data Structures

INSTRUCTOR(S): Marc Liberatore

This course 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, assignments in discussion sections consisting of programming and written exercises, several announced quizzes, and a 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). 4 credits.

COMPSCI 187 Programming with Data Structures

INSTRUCTOR(S): Rui Wang, 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 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 197C Special Topics - Programming in C

INSTRUCTOR(S): Nicolas Scarrci

A brief introduction to the C programming language for students with a good working knowledge of Java and data structures. This course is good preparation for COMPSCI 230 and courses that use C and C++. Prerequisites: COMPSCI 121 and 187. Runs for 6 weeks. This course is for CS minors and majors only, but it does not count towards either degree. 1 credit.

COMPSCI 197S Special Topics - Basics of Cybersecurity

INSTRUCTOR(S): Joshua Pikovsky

This short course provides an introduction to basic principles and practices of computer and network security. Topics include ethics, primary definitions, applied cryptography, networking, operating systems, malware, and incident handling. The course is flipped: students complete readings and assignments ahead of each class, and class time is used to discuss this work. Class discussion includes current events related to security as well. Attendance is mandatory. Open to FROSH/SOPH CS Majors only. 1 credit.

COMPSCI 197U Special Topics - A Hands-on Introduction to UNIX

INSTRUCTOR(S): John Foley

This course offers a 6-week introduction to working with Unix, and it is intended to help students work with tools commonly used in CS courses. The class is comprised of both discussion and hands-on exercises in the EdLab. Topics covered include working with the command line, installing and maintaining the OS and software packages, version control systems, compiling programs, and more. No previous experience with Unix is required. This course is for CS minors and majors only, but it does not count towards either degree. 1 credit.

COMPSCI 197WP Special Topics - Introduction to HTML/CSS

INSTRUCTOR(S): Nicolas Scarrci

An academic approach to building websites with html and css. This course is aimed at beginning programmers and self-taught web developers who would like to cement their skills. Topics include structural html5, css, and introductory javascript/jquery (if time permits). Prerequisite: COMPSCI 121. Mandatory P/F. Students who are enrolled in or have taken COMPSCI 326 are not eligible to take this course. 1 credit.

COMPSCI 198J Practicum - High School Teaching Practicum

INSTRUCTOR(S): Emery Berger, David Barrington

Assisting high school students in learning computer programming in the context of a high school course. The language of instruction will be Java or Python. Students taking this independent study will be expected to work under the direct supervision of the high school computer science teacher at a local high school. Students are expected to attend classes weekly for approximately one hour. Course evaluation will be performed by the high school teacher in cooperation with the supervising faculty member, and a final written report will be required.

COMPSCI 220 Programming Methodology

INSTRUCTOR(S): Gordon Anderson

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 or ECE 242. 4 credits.

COMPSCI 230 Computer Systems Principles

INSTRUCTOR(S): Timothy Richards

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. Prerequisites: COMPSCI 187 or ECE 242. 4 credits.

COMPSCI 240 Reasoning Under Uncertainty

INSTRUCTOR(S): Andrew McGregor

Development of mathematical reasoning skills for problems that involve uncertainty. Each concept will be illustrated by real-world examples and demonstrated though in-class and homework exercises, some of which will involve Java programming. Counting and probability -- basic counting problems, probability definitions, mean, variance, binomial distribution, Markov and Chebyshev bounds. Probabilistic reasoning -- conditional probability and odds, Bayes' Law, Naive Bayes classifiers, Monte Carlo simulation. Markov chains, Markov decision processes, classical game theory, introduction to information theory. Prerequisites: COMPSCI 187 (or ECE 242) and MATH 132 or consent of instructor. 4 credits.

COMPSCI 250 Introduction to Computation

INSTRUCTOR(S): David Barrington

Lecture, discussion. 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, 2-3 midterm exams, timed final. Prerequisite: MATH 132 and COMPSCI 187 (or ECE 242). MATH 132 may be used as a co-requisite with permission of instructor. 4 credits.

COMPSCI 305 Social Issues in Computing

INSTRUCTOR(S): Michelle Trim, Dawn Lovegrove, Anna Napoleone

Through a careful analysis and discussion of a range of computing issues, topics, and polices, 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 equivalent and COMPSCI 220 (or COMPSCI 230) and COMPSCI 240 (or COMPSCI 250). 3 credits.

COMPSCI 311 Introduction to Algorithms

INSTRUCTOR(S): Ramesh Sitaraman, Daniel Sheldon

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 250 or MATH 455. 4 credits.

COMPSCI 320 Introduction to Software Engineering

INSTRUCTOR(S): Rene Just, Neena Thota

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 326 Web Programming

INSTRUCTOR(S): Timothy Richards

The World Wide Web was proposed originally as a collection of static documents inter-connected by hyperlinks. Today, the web has grown into a rich platform, built on a variety of protocols, standards, and programming languages, that aims to replace many of the services traditionally provided by a desktop operating system. Topics will include: producing dynamic content using a server-based language, content serving databases and XML documents, session state management, multi-tier web-based architectures, web security, and core technologies including HTTP, HTML5, CSS, JavaScript, and SQL will be emphasized. This course will also study concepts and technologies including AJAX, social networking, mashups, JavaScript libraries (e.g., jQuery), and web security. This course is hands-on and project-based; students will construct a substantial dynamic web application based on the concepts, technologies, and techniques presented during lecture. This course satisfies the IE Requirement. Prerequisites: COMPSCI 220 or COMPSCI 230. 3 credits.

COMPSCI 348 Principles of Data Science

INSTRUCTOR(S): David Jensen

Knowledge discovery is the process of discovering useful regularities in large and complex data sets. The field encompasses techniques from artificial intelligence (representation and search), statistics (inference), and databases (data storage and access). When integrated in to useful systems, these techniques can help human analysts make sense of vast stores of digital information. This course presents the fundamental principles of the field, 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. Prerequisites: COMPSCI 187 (or ECE 242), COMPSCI 240, and COMPSCI 250. 3 credits.

COMPSCI 365 Digital Forensics

INSTRUCTOR(S): Marc Liberatore

The goal of forensics is to gather artifacts for refinement into evidence that supports or refutes a hypothesis about an alleged crime or policy violation. Done correctly, forensics represents the application of science to law. The techniques can also be abused to thwart privacy. This course is a broad introduction to forensic investigation of digital information and devices. We will cover the acquisition, analysis, and courtroom presentation of information from file systems, operating systems, networks, cell phones, and the like. Students do not need experience with these systems. We will review the use of some professional tools that automate data harvesting, however, the primary goal of the class is to understand why and from where artifacts are recoverable in these systems. Several assignments involve coding forensic tools from scratch. For a small portion of the class, we will cover some relevant issues from the law, privacy, and current events. Thus, the class serves the well-rounded student who is eager to participate in class discussion on a variety of technical and social issues. Prerequisites: COMPSCI 220 or COMPSCI 230; CS majors only. 3 credits.

COMPSCI 370 Introduction to Computer Vision

INSTRUCTOR(S): Subhransu Maji

This introductory computer vision class will address fundamental questions about getting computers to "see" like humans. We investigate questions such as -What is the role of vision in intelligence? -How are images represented in a computer? -How can we write algorithms to recognize an object? -How can humans and computers "learn to see better" from experience? We will write a number of basic computer programs to do things like recognize handwritten characters, track objects in video, and understand the structure of images. Prerequisite: COMPSCI 240 or 383. 3 credits.

COMPSCI 373 Introduction to Computer Graphics

INSTRUCTOR(S): Evangelos Kalogerakis

This course introduces the fundamental concepts of 2D and 3D computer graphics. It covers the basic methods for modeling, rendering, and imaging. Topics include: image processing, digital photography, 2D/3D modeling, 3D graphics pipeline, OpenGL, shading, texture mapping, ray tracing, 3D printing. Throughout the class, we will teach students to learn modern graphics techniques, to model the visual world algorithmically, and to implement algorithms using Java. Students who have taken COMPSCI 473 are not eligible to take this course. Prerequisites: COMPSCI 187 (or ECE 242) and COMPSCI 190DM (or MATH 235 or COMPSCI 240 or equivalent courses from other departments). 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): Kristina Fedorenko, Richard Freedman

The course explores key concepts of artificial intelligence, including state-space and heuristic search techniques, problem-solving, game playing, knowledge representation, automated planning, reasoning under uncertainty, decision theory, and machine learning. AI is largely defined as the study of agents that receive percepts from the environment and perform actions. Each such agent implements a function that maps percept sequences to actions, and the course covers different ways to represent these functions, such as production systems, reactive agents, real-time conditional planners, neural networks, and decision-theoretic systems. Prerequisites: COMPSCI 220 (or COMPSCI 230) and COMPSCI 240. 3 credits.

COMPSCI 390N Internet of Things

INSTRUCTOR(S): Sunghoon Lee

We are living in a world where everyday objects, such as smartphones, cars, TVs, and even refrigerators, are becoming smarter and constantly connected to each other to build, operate, and manage the physical world. This emerging paradigm, namely the Internet of Things (IoT), has great potential to impact how individuals live and work by providing a source of innovative decision making. The design of the IoT, which is defined as “a internetwork of physical items – each embedded with sensors – that are connected to the Internet”, requires the understanding of embedded electronics, software, sensors, network, and data analytics. To prepare our students as forerunners of this future, this course will introduce a wide range of topics in the broad areas of IoT, and provide hands-on experiences via a series of exciting projects. Prerequisite: COMPSCI 187. This course counts as a CS Elective for the CS Major (BA/BS). 3 credits.

COMPSCI 445 Information Systems

INSTRUCTOR(S): Gerome Miklau

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. map-reduce). Additional application topics may include web application development, data integration, processing data streams, database security and privacy. Students who have completed COMPSCI 345 are not eligible to take this course without instructor permission. Prerequisite: COMPSCI 220 (or 230) and COMPSCI 311. 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, or equivalent. 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, applications, network, programming interfaces, transport, congestion, routing, data link protocols, local area networks, emerging high-speed 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. Prerequisites: Experience programming; COMPSCI 230 (or COMPSCI 377). 3 credits.

COMPSCI 491G Seminar - 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 toward the CS major (BA/BS). Prerequisite: COMPSCI 453. 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. Prerequisites: COMPSCI 311 or equivalent. It is recommended that students have a ‘B-’ or better in 311 in order to attempt 501. 3 credits.

COMPSCI 503 Embedded Computing Systems

INSTRUCTOR(S): Roderic Grupen

This course introduces tools for embedded computational applications in a class focused on team-oriented design applications. Your team will build an integrated robot to perform a challenge task and will compete against other teams for the best system. The course is heavily project-oriented (with a required lab) and discussions will include topics such as; (1) mechanisms, sensors, actuators and feedback systems, (2) analog and digital circuits, power amplifiers, signal processing, operational amplifiers, multiplexing, (3) I/O - A/D, D/A, and latching, serial and parallel interfaces, (4) signal processing/conditioning and (5) an introduction to real-time programming. 3 credits.

COMPSCI 521 Software Engineering: Analysis and Evaluation

INSTRUCTOR(S): Yuriy Brun

Software has become ubiquitous in our society. It controls life-critical applications, such as air traffic control and medical devices, and is of central importance in telecommunication and electronic commerce. In this course, we will examine state-of-the-art practices for software testing and analysis to verify software quality. We will initially look at techniques for testing and analyzing sequential programs, and then examine the complexity that arises from distributed programs. The students will be required to complete regular homework assignments and exams, and carry out a group research project extending techniques described in class and/or applying them to new domains. No required text. Papers from the open literature will be assigned and made available. Prerequisites: COMPSCI 320, Introduction to Software Engineering (or equivalent course). 3 credits.

COMPSCI 529 Software Engineering Project Management

INSTRUCTOR(S): Rene Just, Neena Thota

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 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. Weekly assignments, semester project, 2 hours exams, final. Prerequisites: COMPSCI 391IB/335. 3 credits.

COMPSCI 589 Machine Learning

INSTRUCTOR(S): Benjamin Marlin

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 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 also explore the use of machine learning methods across different computing contexts including desktop, cluster, and cloud computing. The course will include programming assignments, a midterm exam, and a final project. Python is the required programming language for the course. Prerequisites: COMPSCI 383 and MATH 235. 3 credits.

COMPSCI 590F Advanced Digital Forensics

INSTRUCTOR(S): Marc Liberatore

This course offers a broad introduction to the forensic investigation of digital devices. We cover the preservation, recovery, harvesting, and courtroom presentation of information from file systems, operating systems, networks, database systems applications, media files, and embedded systems. The primary goal of the class is to understand why and from where information is recoverable in these systems. We also cover relevant issues from criminology, law, and the study of privacy. Students who have completed COMPSCI 365 are not eligible for this course. Graduate students will be expected to read selected current technical and law articles and to implement and analyze state-of-the-art systems. 3 credits.

COMPSCI 590P Secure Distributed Systems

INSTRUCTOR(S): Brian Levine

In this course, we will study digital currencies as a platform for studying the security and privacy needs of real systems, which is increasingly a multidisciplinary task. Even if systems could be "fully secured" against all technical vulnerabilities they would still play a role in violating higher-layer policies. In the context of distributed digital currencies (such as bitcoin), we will examine distributed systems security, blockchain-based consensus, cryptography that supports digital currency, network-based attacks, and the financial, legal, and social issues the technology perturbs. Assignments will include advanced programming projects and reading research papers. This course counts as a CS Elective toward the CS major (BA/BS), as well an Any 2 menu choice for the Security & Privacy track, as well as a Systems Core for the MS. Undergraduate prerequisite: COMPSCI 377, COMPSCI 460, COMPSCI 453, COMPSCI 410, COMPSCI 474, COMPSCI 491IP, or COMPSCI 497P. 3 credits.

COMPSCI 590V Data Visualization and Exploration

INSTRUCTOR(S): Georges Grinstein

In this course, students will learn the fundamental algorithmic and design principles of visualizing and exploring complex data. The course will cover multiple aspects of data presentation including human perception and design theory; algorithms for exploring patterns in data such as topic modeling, clustering, and dimensionality reduction. A wide range of statistical graphics and information visualization techniques will be covered. We will explore numerical data, relational data, temporal data, spatial data, graphs and text. Hands-on projects will be based on Python or JavaScript with D3. This course counts as a CS Elective toward the CS major (BA/BS). Undergraduate Prerequisite: COMPSCI 220 or 230. No prior knowledge of data visualization or exploration is assumed. This course counts as a CS Elective toward the CS major (BA/BS). 3 credits.

COMPSCI 601 Computation Theory

INSTRUCTOR(S): Neil Immerman

An in-depth introduction to the main models and concepts of the theory of computation, including: Computability: what problems can be solved in principle; Complexity: what problems can be solved in a given amount of time, space, parallel time; Logic: how do formal specification and proof mirror other forms of computation? Students will learn to go from a concrete problem to a mathematical model; and, after proving things about the mathematical model, to correctly interpret what they have learned about the concrete problem. Prerequisites: an undergraduate course in automata theory and formal languages such as COMPSCI 501 or permission of instructor. Course requirements: biweekly problem sets, midterm and final. Also open to qualified undergraduates. 3 credits.

COMPSCI 603 Robotics

INSTRUCTOR(S): Joydeep Biswas

This course is intended to serve as an advanced overview of robotics spanning the complete autonomy loop: robot hardware, perception, planning, and control. We will study algorithms and data structures related to these topics, covering widely adopted, and state of the art techniques. Students will gain hands-on experience in implementing, and extending such algorithms using real robot data, as well as simulations. 3 credits.

COMPSCI 621 Advanced Software Engineering: Analysis and Evaluation

INSTRUCTOR(S): Yuriy Brun

Software has become ubiquitous in our society. It controls life-critical applications, such as air traffic control and medical devices, and is of central importance in telecommunication and electronic commerce. In this course, we will examine state-of-the-art practices for software testing and analysis to verify software quality. We will initially look at techniques for testing and analyzing sequential programs, and then examine the complexity that arises from distributed programs. The students will be required to complete regular homework assignments and exams, and carry out a group research project extending techniques described in class and/or applying them to new domains. No required text. Papers from the open literature will be assigned and made available. Prerequisites: COMPSCI 320, Introduction to Software Engineering (or equivalent course). 3 credits.

COMPSCI 630 Systems

INSTRUCTOR(S): Emery Berger

This class is an in-depth introduction to systems, focusing on principles of system design that cross-cut numerous systems artifacts, including operating systems, databases, runtime systems, and architecture. We will cover all levels of the "system stack", from chips to distributed systems. This class may be used to satisfy systems core requirements. 3 credits.

COMPSCI 645 Database Design and Implementation

INSTRUCTOR(S): Alexandra Meliou

This course covers the design and implementation of traditional relational database systems and advanced data management systems. The course will treat fundamental principles of databases: the relational model, conceptual design, query languages, and selected theoretical topics. We also cover core database implementation issues including storage and indexing, query processing and optimization, as well as transaction management, concurrency, and recovery. Additional topics will address the challenges of modern Internet-based data management. These include data mining, provenance, information integration, incomplete and probabilistic databases, and database security. 3 credits.

COMPSCI 660 Advanced Information Assurance

INSTRUCTOR(S): Amir Houmansadr

This course provides an in-depth examination of the fundamental principles of information assurance. While the companion course for undergraduates is focused on practical issues, the syllabus of this course is influenced strictly by the latest research. We will cover a range of topics, including authentication, integrity, confidentiality of distributed systems, network security, malware, privacy, intrusion detection, intellectual property protection, and more. Prerequisites: COMPSCI 460 or 466, or equivalent. 3 credits.

COMPSCI 677 Distributed and Operating Systems

INSTRUCTOR(S): Prashant Shenoy

This course provides an in-depth examination of the principles of distributed systems in general, and distributed operating systems in particular. Covered topics include processes and threads, concurrent programming, distributed interprocess communication, distributed process scheduling, virtualization, distributed file systems, security in distributed systems, distributed middleware and applications such as the web and peer-to-peer systems. Some coverage of operating system principles for multiprocessors will also be included. A brief overview of advanced topics such as multimedia operating systems and mobile computing will be provided, time permitting. *Prerequisites:* Students should be able to easily program in a high-level language such as C, 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. Lect 2 is on-line. 3 credits.

COMPSCI 683 Artificial Intelligence

INSTRUCTOR(S): Shlomo Zilberstein

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 685 Advanced Natural Language Processing

INSTRUCTOR(S): Brendan O'Connor

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. 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, 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 homework assignments and a final project. 3 credits.

COMPSCI 688 Probabilistic Graphical Models

INSTRUCTOR(S): Justin Domke

Probabilistic graphical models are an intuitive visual 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 that commonly occur 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 and their dynamic and relational extensions; (b) exact and approximate inference methods; (c) estimation of both the parameters and structure of graphical models. Although the course is listed as a seminar, it will be taught as a regular lecture course with programming assignments and exams. Students entering the class should have good programming skills and knowledge of algorithms. Undergraduate-level knowledge of probability and statistics is recommended. 3 credits.

COMPSCI 690T Coding Theory and Applications

INSTRUCTOR(S): Arya Mazumdar

Introduction to Coding Theory, Algebraic Codes, Reed-Solomon Codes and List Decoding, Reed-Muller Codes, Random Coding and Asymptotical Goodness, LDPC and Expander Codes, Applications in Inference: Sparse-Graph Codes, Iterative Message-passing Decoding, Applications in Networks and Security: Network Coding, Secret Sharing Schemes. The main prerequisites for this class are mathematical maturity, exposure to basic mathematical courses such as COMPSCI 240 and MATH 235 and a solid grounding in linear algebra and probability theory. For most graduate students, undergraduate preparation in these areas is assumed. Students without such background can seek permission of the instructor. 3 credits.

COMPSCI 691BR Seminar - Building a Robot Soccer Team

INSTRUCTOR(S): Joydeep Biswas

Robot soccer has been adopted as a test bed and progress marker for Artificial Intelligence and Robotics research since 1996. The primary success of soccer as a test bed is that the high-level evaluation of a successful team is readily understandable to everyone: the team that scores more goals, and "plays better soccer", is likely the team with the most effective innovations. The goal of this 3-credit seminar is to explore the different sub-problems of AI and robotics as applied to robot soccer, to survey the state of the art, and to implement the most promising approach from each sub-problem. The expected outcomes of the class is a group understanding of the state of the art in the sub-problems of robot soccer, and to have a codebase with high-quality implementations of the state of the art, which will then be used for the UMass RoboCup team at the 2017 RoboCup competition at Nagoya, Japan. The league that we will be entering in is the Small Size League (SSL). Rules, descriptions, and team description papers of teams from past RoboCup competitions are available on the RoboCup SSL Wiki. 3 credits.

COMPSCI 691RS Seminar - Introduction to Recommender Systems

INSTRUCTOR(S): Yongfeng Zhang

Recommender Systems (RS) are key technical components in many online applications, such as product recommendation, video recommendation, or stream recommendation in social networks. In this course, we cover the applications and technical details of recommender systems, including: 1) Historical overview 2) Major recommendation frameworks, including content-based, collaborative filtering, and hybrid recommender systems; 3) Current techniques for recommendation, including matrix factorization, frequent pattern mining, sentiment analysis, deep learning, and economic models; 4) Analysis of recommendation applications. From this course, students will develop an understanding of the research issues, technical basis, and practical applications of personalized recommender systems. 3 credits.

COMPSCI 697W Special Topics - Wearable and Mobile Sensor Computing

INSTRUCTOR(S): Deepak Ganesan

The ubiquity of sensor-equipped smartphones and wearables has led to increased interest in leveraging these signals for mobile healthcare, wearability and interaction, entertainment, and other applications. This seminar will explore research issues in sensor computing including: a) inference of human activity, gestures, cognitive state, health state, and physiological markers, by leveraging sensor signals from wearable devices, mobile phones, radio-frequency devices, and other sources, b) computational pipelines for extracting features and analyzing data from a variety of sensors (e.g. accelerometer, gyroscope, microphone, camera, radio, etc), and c) end-to-end systems design from physical layer to end applications.

COMPSCI 701 Advanced Topics in Computer Science

INSTRUCTOR(S):

This is a 6 credit reading course corresponding to the master's project. The official instructor is the GPD although the student does the work with and is evaluated by the readers of his or her master's project.

COMPSCI 891M Seminar - Theory of Computation

INSTRUCTOR(S): David Barrington

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 H445 Honors Colloquium for Information Systems

INSTRUCTOR(S): Gerome Miklau

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. map-reduce). Additional application topics may include web application development, data integration, processing data streams, database security and privacy. The colloquium will focus on advanced topics and recent research topics related to information management and data science. Students will participate in group discussions and carry out a group or individual project which will extend the project work in COMPSCI 445. Students will be graded based on their active participation during meetings, written summaries of assigned readings, and project work. Prerequisite: COMPSCI 220 (or 230) and COMPSCI 311. 4 credits.

COMPSCI H589 Honors Colloquium for Machine Learning

INSTRUCTOR(S): Benjamin Marlin

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 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 also explore the use of machine learning methods across different computing contexts including desktop, cluster, and cloud computing. The course will include programming assignments, a midterm exam, and a final project. Python is the required programming language for the course. This course includes an honors colloquium for COMPSCI 589: Machine Learning. It will include an exploration of the mathematical foundations of the machine learning algorithms presented in class. It will also include the presentation of more advanced models and algorithms. Students will participate in and lead group discussions on these topics, as well as on their final course projects. Prerequisites: COMPSCI 383 and MATH 235. 4 credits.

INFO 203 A Networked World

INSTRUCTOR(S): Phillipa Gill

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.

INFO 248 Introduction to Data Science

INSTRUCTOR(S): Gordon Anderson

The terms "data science" and "big data" appear in the news media and in everyday conversations. Moreover, we are told that we live in the "age of information", where almost every business venture and scientific research initiative collect a massive amount of data which may contain valuable information. This course is an introduction to the concepts and skills involved with the collection, management, analysis, and presentation of large data sets and the data products that result from the work of data scientists. Privacy and ethical issues are discussed. Students will work with data from the financial, epidemiological, educational, and other domains.   The course provides many case studies and examples of real-world data that students work with using the R programming language as well as the structured query language (SQL). This course consists of two meetings per week. Each meeting includes a lecture, where conceptual material will be presented, followed by lab time where students receive instruction on the use of software tools and apply the concepts by working on data sets. Readings will be assigned as preparation for each class meeting. Several projects will be assigned during the course. The projects provide students with an opportunity to explore the topics in more depth in a specialized domain. Two midterm exams and one final exam will be given. Grades are determined by a combination of class participation including the in-class lab activities, projects, and exam scores. Software: The R software for statistical analysis (www.r-project.org). This course does not satisfy requirements for the CS major. Prerequisites: COMPSCI 190IN and (COMPSCI 119 or COMPSCI 121). 3 credits.

INFO 397G Special Topics - Creative Game Design and Development

INSTRUCTOR(S): Joshua Newman

In this class, we will explore, through a series of projects, the fundamental questions of game design. What are the common features of hopscotch, Skyrim, boxing, Farmville, poker, and Tic-Tac-Toe? How do you create an engrossing, challenging, vivid, or surprising environment of play? How do you determine the value of skill, chance, cooperation, and competition in game play? What effect does the social, sexual, gender, political, and economic environment of the game's creation have on the play of the game? This course will introduce students to theoretical structures and demonstrate their implementation through the development of several small physical and digital games, ending with a group digital game project. Students will be evaluated based on class participation, process papers, and the creation of their own games. Not for CS Major/Minor Requirements. 3 credits.