

**COMPSCI 105 Computer Literacy**

3 Credits

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. 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**

3 Credits

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**

4 Credits

INSTRUCTOR(S): Neena Thota, Gordon Anderson

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

**COMPSCI 145 Representing, Storing, and Retrieving Information**

3 Credits

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 heterogeneous 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**

4 Credits

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. Was COMPSCI 190D. Prerequisites: COMPSCI 121 and Basic Math Skills (R1). 4 credits.

---

**COMPSCI 187 Programming with Data Structures**

4 Credits

INSTRUCTOR(S): Gordon Anderson, Meng-Chieh Chiu

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

**COMPSCI 197C Special Topics - Programming in C**

1 Credits

INSTRUCTOR(S): Srideepika Jayaraman

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 186 or 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 197U Special Topics - A Hands-on Introduction to UNIX**

1 Credits

INSTRUCTOR(S): Arun Dunna

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 198J Practicum - High School Teaching Practicum**

1 Credits

INSTRUCTOR(S): Ramesh Sitaraman

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. In some terms students are expected to attend high school classes weekly for approximately one hour. In others they will work on developing projects for future courses, meeting with the high school instructor once a week. 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. 1 credit.

**COMPSCI 220 Programming Methodology**

4 Credits

INSTRUCTOR(S): Arjun Guha, Meng-Chieh Chiu

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

**COMPSCI 230 Computer Systems Principles**

4 Credits

INSTRUCTOR(S): Timothy Richards, Nic Herndon

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. 4 credits.

**COMPSCI 240 Reasoning Under Uncertainty**

4 Credits

INSTRUCTOR(S): Nic Herndon, Shiting Lan

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

---

**COMPSCI 250 Introduction to Computation**

4 Credits

INSTRUCTOR(S): Marius Minea, Swarna Reddy

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 midterm exams, timed final. Prerequisite: MATH 132 and COMPSCI 187. 4 credits.

**COMPSCI 305 Social Issues in Computing**

3 Credits

INSTRUCTOR(S): Michelle Trim, Justin Obara

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

**COMPSCI 311 Introduction to Algorithms**

4 Credits

INSTRUCTOR(S): Daniel Sheldon, Marius Minea

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

**COMPSCI 320 Introduction to Software Engineering**

4 Credits

INSTRUCTOR(S): David Fisher, 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**

4 Credits

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. 4 credits.

**COMPSCI 345 Practice and Applications of Data Management**

3 Credits

INSTRUCTOR(S): Alexandra Meliou

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

---

**COMPSCI 348 Principles of Data Science**

3 Credits

INSTRUCTOR(S): David Jensen

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. Prerequisites: COMPSCI 187, COMPSCI 240, and COMPSCI 250 (or MATH 455). 3 credits.

**COMPSCI 365 Digital Forensics**

3 Credits

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 230; CS majors only. 3 credits.

**COMPSCI 370 Introduction to Computer Vision**

3 Credits

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**

3 Credits

INSTRUCTOR(S): Rui Wang

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. Students cannot take COMPSCI 497C after taking this course. Prerequisites: COMPSCI 187 and INFO 150 (or MATH 235 or COMPSCI 240). 3 credits.

**COMPSCI 377 Operating Systems**

4 Credits

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**

3 Credits

INSTRUCTOR(S): Matthew Rattigan

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

---

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

3 Credits

INSTRUCTOR(S): Roderic Grupen

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

**COMPSCI 446 Search Engines**

3 Credits

INSTRUCTOR(S): David Fisher

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

**COMPSCI 453 Computer Networks**

3 Credits

INSTRUCTOR(S): Arun Venkataramani

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 466 Applied Cryptography**

3 Credits

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. Prerequisites: COMPSCI 311. 3 credits.

**COMPSCI 490P Secure Distributed Systems**

3 Credits

INSTRUCTOR(S): Brian Levine

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

**COMPSCI 490S Software Entrepreneurship**

3 Credits

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. Prerequisites: COMPSCI 320 (or COMPSCI 320). This course counts as a CS Elective toward the CS major (BA or BS). 3 Credits.

---

**COMPSCI 491G Seminar - Computer Networking Lab**

3 Credits

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**

3 Credits

INSTRUCTOR(S): Marius Minea

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. Undergraduate 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 514 Algorithms for Data Science**

3 Credits

INSTRUCTOR(S): Barna Saha

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. Course was previously COMPSCI 590D. Undergraduate Prerequisites: COMPSCI 240 and COMPSCI 311. 3 credits

**COMPSCI 529 Software Engineering Project Management**

3 Credits

INSTRUCTOR(S): David Fisher, 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**

3 Credits

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. Undergraduate Prerequisites: COMPSCI 335. 3 credits.

**COMPSCI 574 Intelligent Visual Computing**

3 Credits

INSTRUCTOR(S): Evangelos Kalogerakis

Intelligent visual computing is an emerging new field that seeks to combine modern trends in machine learning, computer graphics, computer vision to intelligently process, analyze and synthesize 2D/3D visual data. The course will start by covering 2D image and 3D shape representations, classification and regression techniques, and the fundamentals of deep learning. The course will then provide an in-depth background on analysis and synthesis of images and shapes with deep learning, in particular convolutional neural networks, recurrent neural networks, memory networks, auto-encoders, adversarial networks, reinforcement learning methods, and probabilistic graphical models. Students will complete 5 programming assignments in Matlab/Octave and work on a course project related to visual computing with machine learning. This course counts as a CS Elective toward the CS major (BA/BS). Course was previously COMPSCI 590IV. Undergraduate Prerequisites: COMPSCI 311, COMPSCI 383, COMPSCI 373 (or COMPSCI 473). 3 credits.

---

**COMPSCI 589 Machine Learning**

3 Credits

INSTRUCTOR(S): Justin Domke, Swarna Reddy

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. Students will complete programming assignments and exams. Python is the required programming language for the course. Undergraduate Prerequisites: COMPSCI 383 and MATH 235. 3 credits.

**COMPSCI 590F Advanced Digital Forensics**

3 Credits

INSTRUCTOR(S): Marc Liberatore

This course introduces students to the principal activities and state-of-the-art techniques involved in developing digital forensics systems. Topics covered may include: advanced file carving and reconstruction, forensic analysis of modern filesystems, network forensics, mobile device forensics, memory forensics, and anti-forensics. This course counts as a CS Elective toward the CS major (BA or BS). 3 credits.

**COMPSCI 590G Game Programming**

3 Credits

INSTRUCTOR(S): Evangelos Kalogerakis

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

**COMPSCI 590U Mobile and Ubiquitous Computing**

3 Credits

INSTRUCTOR(S): Tauhidur Rahman

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. Undergraduate Prerequisites: COMPSCI 230 and COMPSCI 240. This course counts as a CS Elective toward the CS major (BA or BS). 3 credits.

**COMPSCI 590V Data Visualization and Exploration**

3 Credits

INSTRUCTOR(S): Ali Sarvghad Batn Moghaddam

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 or BS). 3 credits.

**COMPSCI 591A Seminar- Cellular Automata and Artificial Life**

1 Credits

INSTRUCTOR(S): Edward Rietman

Students will present a paper from readings on cellular automata and artificial life. There will be at least two guest speakers. Topics include, but not limited to: cellular automata as computational engine, artificial chemistries, self-assembly, reconfigurable robotics, collective behaviors of swarm, biological computation other than neural networks, systems and synthetic biology, etc. This course does not count toward any requirements for the CS major or minor. 1 credit.

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

3 Credits

INSTRUCTOR(S): Parviz Kermani

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

---

**COMPSCI 601 Computation Theory**

3 Credits

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**

3 Credits

INSTRUCTOR(S): Joydeep Biswas

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 a 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 real robot data, as well as simulations. 3 credits.

**COMPSCI 611 Advanced Algorithms**

3 Credits

INSTRUCTOR(S): Ramesh Sitaraman

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

**COMPSCI 660 Advanced Information Assurance**

3 Credits

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 661 Secure Distributed Systems**

3 Credits

INSTRUCTOR(S): Brian Levine

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

**COMPSCI 674 Intelligent Visual Computing**

3 Credits

INSTRUCTOR(S): Evangelos Kalogerakis

Intelligent visual computing is an emerging new field that seeks to combine modern trends in machine learning, computer graphics, computer vision to intelligently process, analyze and synthesize 2D/3D visual data. The course will start by covering 2D image and 3D shape representations, classification and regression techniques, and the fundamentals of deep learning. The course will then provide an in-depth background on analysis and synthesis of images and shapes with deep learning, in particular convolutional neural networks, recurrent neural networks, memory networks, auto-encoders, adversarial networks, reinforcement learning methods, and probabilistic graphical models. Students will complete 5 programming assignments in Matlab/Octave and work on a course project related to visual computing with machine learning. Course was previously COMPSCI 690IV. 3 credits.

---

**COMPSCI 677 Distributed and Operating Systems**

3 Credits

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**

3 Credits

INSTRUCTOR(S): Madalina Fiterau Brostean

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**

3 Credits

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 690A Advanced Methods in HCI**

3 Credits

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 690D Deep Learning for Natural Language Processing**

3 Credits

INSTRUCTOR(S): Mohit Iyyer, Brendan O'Connor

This course offers an introduction to the models and principles behind state-of-the-art deep learning techniques applied to natural language processing problems. It is intended for graduate students in computer science and linguistics who are (1) interested in learning about cutting-edge research progress in NLP and (2) familiar with machine learning fundamentals. We will cover a variety of models, including vector-based word representations, basic neural network architectures (e.g., convolutional, recurrent), and more advanced variants of these networks that are especially useful for NLP (e.g., attention-based or memory-augmented). We will also see these models in action on a variety of NLP tasks, including text classification, question answering, and text generation. Coursework includes reading recent research papers, programming assignments, and a final project. 3 credits.

**COMPSCI 690OP Optimization in Computer Science**

3 Credits

INSTRUCTOR(S): Arya Mazumdar

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. There is currently no course in the department that covers these topics, and yet it is critical to a large number of research projects done within the department. 3 credits.

---

**COMPSCI 690W Advanced Wireless Networking and Sensing in IoT**

3 Credits

INSTRUCTOR(S): Jie Xiong

Nowadays, wireless technologies (such as 802.11 Wi-Fi) do not only provide data service but also cater to diverse applications including indoor localization, contactless activity sensing, medical implant tracking and charging, virtual reality (VR) and autonomous cars etc. this course introduces the students with fundamentals in wireless networking and also the state-of-the-art sensing applications in the Era of Internet-of-Things. Wireless sensing is also an active research area which involves wireless communication, signal processing, human computer interaction, machine learning and also hardware prototyping. 3 credits.

**COMPSCI 691M Seminar - Adversarial Machine Learning**

1 Credits

INSTRUCTOR(S): Brian Levine, David Jensen

Adversarial machine learning is the study of the attacks that systematically alter the inferences produced learned models. ML systems are increasingly foundational for large technological systems such as social media, news recommendation, fraud detection, autonomous vehicles, and systems security. In this course, we will read and discuss central papers from a variety of machine learning and security conferences and journals. Students will be expected read assigned papers ahead of class meetings, to lead discussion for at least one paper during the semester, and to actively participate in all discussions. 1 credit.

**COMPSCI 691PL Seminar - Advanced Programming Languages**

3 Credits

INSTRUCTOR(S): Arjun Guha

This seminar is an introduction to programming and theorem-proving with dependent types. We will use Idris, which is a dependently-typed language that is similar to languages such as Coq and Agda. In the first part of the course, we will individually work through the book Type-Driven Development with Idris. In the latter part, we will undertake a larger project as a group. 3 credits.

**COMPSCI 692D Seminar- Machine Learning and Data Analysis for Networked Systems**

1 - 3 Credits

INSTRUCTOR(S): Ramesh Sitaraman, Donald Towsley

Networked systems are increasingly complex and require sophisticated techniques for monitoring, modeling, deployment, and operations. Powerful tools for data analysis and machine learning show great promise in enhancing the efficiency, performance, and security of networked systems. In this seminar, we will explore topics such as:

- Statistical inference and ML for time series analysis of network data
- ML for distributed network operations, such as caching and load balancing
- Statistical inference and learning to distinguish between normal and attack traffic for DDoS protection
- Learning for congestion control, and video bitrate adaptation
- Statistical inference and ML for energy procurement in data centers

Students can register for this course for either one credit or three credits. All students will be expected to make at least one presentation for a minimum of one credit. In addition, students taking this course for three credits will propose and complete a course project in consultation with the instructors. 1-3 credits.

**COMPSCI 701 Advanced Topics in Computer Science**

6 Credits

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. 6 credits.

**COMPSCI 891M Seminar - Theory of Computation**

1 Credits

INSTRUCTOR(S): Arya Mazumdar

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 H377 Honors Colloquium for Operating Systems**

1 Credits

INSTRUCTOR(S): Mark Corner

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

---

**INFO 203 A Networked World**

3 Credits

INSTRUCTOR(S): David Wemhoener

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**

4 Credits

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 data sets and the data products that result from the work of data scientists. Privacy, algorithmic bias 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 various tools including 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. A project will be assigned during the course. The project provides students with an opportunity to explore the topics in more depth in a specialized domain. Two midterm and final exams 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](http://www.r-project.org)). This course does not satisfy requirements for the CS major. Course was previously INFO 397F. Prerequisites: COMPSCI 121 and PSYCH 240 (or OIM 240, or STAT 240, or STAT 515, or RES ECON 212, or SOCIOL 212) both with a C or better. 4 credits.

**INFO 397G Special Topics - Creative Game Design and Development**

3 Credits

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.