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 CMPSCI 191P or CMPSCI 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 CMPSCI majors. 3 credits.

** DELETED **

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

**CMPSCI 191P**  Introduction to Programming  
*William Verts*

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 CMPSCI 230 and courses that use C and C++. Prerequisites: CMPSCI 121 and 187 or permission of instructor. Runs for 6 weeks. This course is for CMPSCI minors and majors only, but it does not count towards either degree. 1 credit.

**CMPSCI 197JA**  Special Topics - Programming Effectively with Java  
*Ryan Hurley*

When performance and reliability are needed, more and more companies rely on Java. This 6 weeks course will leave students with a deeper knowledge of Java's strengths and weaknesses and a rough idea of what an industry experience might entail. Course topics will include a deep look into different features of Java, common interview questions, patching software, and code reviews. Prerequisites: CMPSCI 187. Runs for 6 weeks. This course is for CMPSCI minors and majors only, but it does not count towards either degree. 1 credit.

**CMPSCI 197U**  Special Topics - A Hands-on Introduction to UNIX  
*Jarred Devaughn-Brown*

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.

**CMPSCI 220**  Programming Methodology  
*J. Eliot Moss, Timothy Richards*

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: CMPSCI 187 or ECE 242. 4 credits.

**CMPSCI 230**  Computer Systems Principles  
*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 the relevant features of modern architectures, and moves up the stack from there to operating system services such as programming language runtime systems, concurrency and synchronization, with a focus on key operating system features, I/O and networking, and distributed services. This course can be used as a "core" course for students in the CMPSCI minor. Prerequisites: CMPSCI 187 or ECE 242. 4 credits.

**CMPSCI 240**  Reasoning Under Uncertainty  
*Andrew McGregor*

MRST 220Development 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. This is a core course for the new CMPSCI curriculum and may be used as a math elective for the old curriculum. Prerequisites: CMPSCI 187 (or ECE 242) and MATH 132 or consent of instructor. 4 credits.
**CMPSCI 250**  
*Introduction to Computation*  
*Neil Immerman, David Barrington*


**CMPSCI 305**  
*Social Issues in Computing*  
*Stephen Constantine*

Designed to satisfy the Junior Year writing requirement, CMPSCI 305 introduces the student to technical writing and editing, scientific journalism and the social essay. The course combines practical, scientific writing as found in industry and business with explorative essays that focus attention upon the technological and humanistic concerns inherent in society. Ten written assignments—two longer papers, eight shorter ones and one oral presentation. Prerequisite: ENGLWP 112 or CW. 3 credits.

**CMPSCI 311**  
*Introduction to Algorithms*  
*Hava Siegelmann*

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 CMPSCI 250) is required. You should also be able to program in Java, C, or some other closely related language. Prerequisite: CMPSCI 250 or MATH 455. 4 credits.

**CMPSCI 320**  
*Introduction to Software Engineering*  
*Yuriy Brun*  

Honors Colloq

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, exams, and a term project. Prerequisite: CMPSCI 220. 4 credits.

**CMPSCI 326**  
*Web Programming*  
*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. Prerequisites: CMPSCI 187 or ECE 242. 3 credits.

**CMPSCI 377**  
*Operating Systems*  
*Deepak Ganesan*

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: CMPSCI 230 with a grade of ‘C’ or better. 4 credits.
CMPSCI 383  Artificial Intelligence
Shlomo Zilberstein
Honors Colloq
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 how these concepts are applied in the context of several applications.
Prerequisites: (CMPSCI 220 or CMPSCI 230) and CMPSCI 240.  3 credits.

CMPSCI 390DB  Practice and Applications of Data Management
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 alternate between lecture and practice, and the students will experience the covered topics through a semester-long collaborative mini-project. This course counts as a CS Elective toward the CMPSCI major (BA/BS). Students who have taken CMPSCI 445 are not eligible to take this course.  3 credits.

CMPSCI 401  Formal Language Theory
David Barrington
Honors Colloq
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: CMPSCI 311 or equivalent. It is recommended that students have a 'B-' or better in 311 in order to attempt 401.  3 credits.

CMPSCI 445  Information Systems
Yanlei Diao
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. Prerequisite: CMPSCI 220 (or 230) and CMPSCI 311.  3 credits.

CMPSCI 446  Search Engines
Mostafa Keikha
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 CMPSCI 240) is required. You should also be able to program in Java (or some other closely related language). This course may be used in place of CMPSCI 445 as a semi-required course under the OLD rules for the BS. Prerequisite: CMPSCI 240 or CMPSCI 383, or equivalent.  3 credits.

CMPSCI 453  Computer Networks
Arunkumar Venkataramani
This course provides an introduction to fundamental concepts in computer networks, including their design and implementation. Topics covered include the Web and other applications, transport protocols (providing reliability and congestion control), routing, and link access. Special attention is also paid to wireless networks and security. Homework assignments involve programming and written tasks. Prerequisites: Experience programming; CMPSCI 230 (or CMPSCI 377) and CMPSCI 240 (or STATS 515). 3 credits.

CMPSCI 473  Introduction to Computer Graphics
Rui Wang
This course teaches the fundamentals of 2D and 3D computer graphics. Students will learn OpenGL and standard graphics algorithms. Specific topics covered include image processing, modeling, rendering, illumination, shadows, textures, and programmable shaders. The completion of this course will prepare students ready for any advanced computer graphics course. Students are expected to be familiar with basic linear algebra (vector and matrix arithmetic) and Java programming. Prerequisites: CMPSCI 187 (or ECE 242) and MATH 235.  3 credits.
CMPSCI 491DD  Seminar - Research Methods for Honors Students

This course introduces to commonwealth honors completing department capstones basic ideas about conducting a personal research program. Students will learn basic methods for activities such as reading technical papers, selecting research topics, devising research questions, planning research, analyzing experimental results, modeling and simulating computational phenomena, and synthesizing broader theories. The course will be structured around three activities: lectures on basic concepts of research strategy and techniques, discussions of technical papers, and preparation and review of written assignments, all centered on the student's specific capstone project. Significant reading, reviewing, and writing will be required, and students will be expected to participate actively in class discussions. 3 credits.

CMPSCI 491G  Seminar - Computer Networking Lab
Arunkumar Venkataramani
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 CMPSCI major (BA/BS). Prerequisite: CMPSCI 453. 3 credits.

CMPSCI 491IP  Seminar - Programming the iPhone and iPad
Charles Weems
The iPad, iPhone, and iPod Touch present examples of a mobile processor with an interesting set of peripheral devices and limitations. They are programmed using Objective C and a large set of APIs. This seminar will be an introduction to the Objective C language and iOS development environment, and will include student programming assignments and development projects targeting the Apple iPhone/iPad simulator. We will spend much of the class time actually working through examples, and discovering how some of the features really work. Students will need to have access to an Intel machine running OSX, because the development environment and simulator work only on that platform. Students also must join the Apple Developer Program, and accept their license agreements in order to obtain and use the necessary software. This course counts as a CS Elective toward the CMPSCI major (BA/BS). 3 credits.

CMPSCI 520  Software Engineering: Synthesis and Development
Leon Osterweil
Introduces students to the principal activities involved in developing high-quality software systems in a variety of application domains. Topics include: requirements analysis, formal and informal specification methods, process definition, software design, testing, and risk management. The course will pay particular attention to differences in software development approaches in different contexts. Prerequisites: CMPSCI 320 with a grade of 'C' or better, or E&C-ENG 373 with a grade of 'C' or better. 3 credits.

CMPSCI 529  Software Engineering Project Management
Yuriy Brun
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 CMPSCI 320. As project managers, the students in CMPSCI 529 will be responsible for: supervising and managing the work of teams of CMPSCI 320 students; interfacing with the other CMPSCI 529 students managing other teams in the course; interfacing with the course instructor, course TA, and course customer. CMPSCI 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, CMPSCI 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. CMPSCI 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 CMPSCI 320. Additional meetings with team members and other students in CMPSCI 529 are also expected to be arranged by mutual agreement. An additional one hour weekly meeting of all of the students in CMPSCI 529 is required. Enrollment in this course is only by permission of the instructor, and is restricted to students who have previously taken CMPSCI 320, and received a grade of A or A-. 3 credits.
CMPSCI 590GM    Geometric Modeling
Evangelos Kalogerakis

Geometric modeling is fundamental to create content for virtual worlds, computer games, computer-animated films, computer-aided design, architecture, multimedia and other related fields. This course will teach students the basics of 3D geometric modeling. It will start by covering the most commonly used digital representations of shapes, such as polygon meshes, point clouds, NURBS surfaces, and subdivision surfaces. The course will also provide an overview of how these digital representations can be created with commercial 3D modeling packages, such as Maya. Then methods for surface parametrization and texturing that add images and detail to shapes will be presented. Methods for acquiring real-world shapes using 3D sensors will also be covered, followed by a description of algorithms for surface reconstruction, denoising and simplification. The course will proceed with an overview of state-of-the-art techniques for shape creation, editing, and deformation. There will be particular focus on techniques for assembly-based modeling, sketch-based modeling, interactive shape editing with curves, and procedural modeling. Finally, probabilistic models for representing shape variability and shape understanding will be presented. Students will be asked to complete two short assignments that require small implementations of geometry processing techniques. There will also be a final project that involves the implementation of a small research idea broadly related to 3D geometric modeling. Towards the mid and end of the course, students will also present papers on state-of-the-art methods for 3D modeling. The marking scheme is 30% assignments, 40% final project, 20% class participation and presentations, 10% reaction reports. A reaction report is a one page "reaction" to one of the papers that is to be discussed that day. The first half of the report should contain discussion of one thing you liked or disliked about the paper. The second half should suggest an idea for "future work:" a possible extension or improvement to the paper. Prerequisites for Undergraduate Students: CMPSCI 240 and CMPSCI 311. There are no prerequisites for CMPSCI graduate students, although familiarity with computer graphics is useful (as covered in CMPSCI 473). Permission of instructor for non-CMPSCI graduate students. This course counts as a CS Elective toward the CMPSCI major (BA/BS). 3 credits.

CMPSCI 601    Computation Theory
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: Undergraduate-level courses in discrete mathematics (CMPSCI 250) and algorithms (CMPSCI 311). Additional mathematical maturity (e.g., CMPSCI 611, A's in 250 or 311, or other mathematical background) quite desirable as is background in automata theory and formal languages (CMPSCI 401). Course requirements: biweekly problem sets, midterm and final. Also open to qualified undergraduates. 3 credits.

CMPSCI 603    Robotics
Roderic Grupen

This course is designed to be a advanced course in robotics that covers mechanisms (kinematics and dynamics), actuators, sensors, signal processing, feedback control, and signal processing. The target is to provide an understanding of robot systems that interact with, interpret feedback from, and manipulate the world about them. We will relate the subject matter to biological systems whenever possible, including discussion about the relationships between learning and development in human beings and what it has to say about programming robots. Students will experiment with system identification and control, image processing, path planning, grasping, and machine learning to reinforce the material covered in class. 3 credits.

CMPSCI 620    Advanced Software Engineering: Synthesis and Development
Leon Osterweil

This course examines the varied approaches to the development of computer software. We examine various ideas about how software products should be structured and function. We then examine how software processes serve as vehicles for manufacturing such products. This approach facilitates the direct study of different software development approaches, and a more direct study of their effects on the products they produce. This approach will be used by students, who will examine representative current software development product and process approaches in in-class presentations and written project papers as part of their coursework. 3 credits.

CMPSCI 635    Modern Computer Architecture
Charles Weems

This course examines the structure of modern computer systems. We explore hardware and technology trends that have led to current machine organizations, then consider specific features and their impact on software and performance. These may include superscalar issue, caches, pipelines, branch prediction, and parallelism. Midterm and final exams, individual projects, homework, in-class exercises. Prerequisites: CMPSCI 535 or equivalent. 3 credits.
CMPSCI 677  Operating Systems
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., CMPSCI 377) will be helpful but not required. Lect 2 is on-line. 3 credits.

CMPSCI 688  Probabilistic Graphical Models
Benjamin Marlin
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.

CMPSCI 690OP  Optimization in Computer Science
Sridhar Mahadevan
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.

CMPSCI 691DD  Seminar - Research Methods in Empirical Computer Science
David Jensen
This course introduces graduate students to basic ideas about conducting a personal research program. Students will learn basic methods for activities such as reading technical papers, selecting research topics, devising research questions, planning research, analyzing experimental results, modeling and simulating computational phenomena, and synthesizing broader theories. The course will be structured around three activities: lectures on basic concepts of research strategy and techniques, discussions of technical papers, and preparation and review of written assignments. Significant reading, reviewing, and writing will be required, and students will be expected to participate actively in class discussions. 3 credits.

CMPSCI 691NN  Seminar - Hot Topics in Programming Languages and Systems
Emery Berger
This graduate seminar course will cover recent developments in programming languages and systems, examining the latest research papers from top programming languages and systems conferences. Topics of interest include bug detection and correction, domain-specific languages, and emerging topics like concurrency on multicore architectures. Lect A=3 credits; Lect B=1 credit.

CMPSCI 691SM  Seminar - Smart Grids
James Kurose, Prashant Shenoy, Krithivasan Ramamritham
The "smart grid" uses computing and communication technologies to enhance the efficiency, reliability, sustainability and flexibility of the generation, distribution and use of electricity. This seminar will introduce participants to the grid and to the smart grid, and will read current smart grid research papers. Topics will include smart grid communication protocols, security and privacy, smart grid monitoring and measurement, demand response, storage, intermittent generation (e.g., renewable sources), smart homes and building and smart grid modeling. This seminar is being offered jointly and collaboratively at The University of Massachusetts Amherst and at The Indian Institute of Technology Bombay. Students may register for 1 credit or 3 credits. Students registering for a single credit will be expected to make a class presentation (developed in consultation with one of the professors) and to actively participate in class discussions. Students registering for 3 credits will also define and complete a research project on a topic of interest, in consultation with the course professors. Projects involving students from both UMass and IITB are particularly encouraged. Pre-requisites: Graduate student status or permission of the instructors. An undergraduate course in operating systems and/or computer networks is required. Lect A=3 credits; Lect B=1 credit.
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.

**CMPSCI 701 Advanced Topics in Computer Science**

STAFF

This seminar will focus on models and algorithms for supervised and unsupervised machine learning with time series. Topics will include discrete and continuous time models from machine learning, statistics and econometrics. We will investigate a variety of time series problems including prediction, detection, clustering, and similarity search. Coursework for the one credit option will include paper presentations and quizzes. Students in the three credit option will also complete a course project. Prerequisites include a graduate-level course in machine learning or graphical models, i.e., CMPSCI 689 or CMPSCI 691GM. Lect A=3 cr; Lect B=1 cr.

**CMPSCI 791TS Seminar - Machine Learning and Time Series**

Benjamin Marlin

This seminar will focus on models and algorithms for supervised and unsupervised machine learning with time series. Topics will include discrete and continuous time models from machine learning, statistics and econometrics. We will investigate a variety of time series problems including prediction, detection, clustering, and similarity search. Coursework for the one credit option will include paper presentations and quizzes. Students in the three credit option will also complete a course project. Prerequisites include a graduate-level course in machine learning or graphical models, i.e., CMPSCI 689 or CMPSCI 691GM. Lect A=3 cr; Lect B=1 cr.

**CMPSCI 891M Theory of Computation**

Andrew McGregor

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.

**CMPSCI H320 Honors Colloquium for CMPSCI 320**

Yuriy Brun

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

**CMPSCI H383 Honors Colloquium for CMPSCI 383**

Shlomo Zilberstein

Students will study in greater depth one of the AI problem solving techniques covered in class. Activities include discussions with the professor and TA, readings, and a project. Specific topics depend on student interest but may include game playing, heuristic search, and automated planning. 1 credit.

**CMPSCI H401 Honors Colloquium for CMPSCI 401**

David Barrington

Students will study additional topics in computability and complexity, through discussions (1 hour/week with professor), readings, and exercises. Specific topics will depend on student interest but may include algebraic automata theory, alternate models of computability, and interactive proofs. 1 credit.