***Comparative Literature 399. Reading and Research**
Catalog Number: 2893
Svetlana Boym 1926, Joaquim-Francisco Coelho 7715, James Engell 8076, Luis M. Girón Negrón 3060, John T. Hamilton 3977 (on leave 2005-06), Christopher D. Johnson 4301, Despina Kakoudaki 3979, Sandra Naddaff 7779, Gregory Nagy 1423, Stephen Owen 7418, Judith Ryan 1135 (on leave fall term), Marc Shell 3176, Werner Sollors 7424, Diana Sorensen 4214, Susan R. Suleiman 7234, William Mills Todd III 1634 (on leave 2005-06), and Ruth R. Wisse 3177
Candidates for the doctoral degree in Comparative Literature may pursue advanced studies under the individual supervision of these instructors.
Note: Permission of the instructor and the Chairman of the Department required.

Cross-listed Courses

***English 251. Comparative Romantic Theory: Graduate Seminar**
***English 292. Issues in the Study of American Literature: Graduate Seminar**
German 234. Kant's *Critique of Judgment* and the Aesthetics of Early German Idealism: Seminar
***Literature 119. Comparative Arts**
Modern Greek 145 (formerly Comparative Literature 145). Dreams and Literature
Visual and Environmental Studies 171h. Histories of Cinema I: Moving Pictures from the 1890s to the 1930s

Computer Science

AN HISTORICAL EDITION OF FAS COURSES OF INSTRUCTION

Faculty of the Division of Engineering and Applied Sciences Offering Instruction in Computer Science

David M. Brooks, Assistant Professor of Computer Science on the Gordon McKay Endowment *(Co-Director of Undergraduate Studies)*
Steven J. Gortler, Robert I. Goldman Professor of Computer Science *(Co-Director of Undergraduate Studies)*
Barbara J. Grosz, Higgins Professor of Natural Sciences
Dan Gutfreund, Lecturer on Applied Mathematics
H. T. Kung, William H. Gates Professor of Computer Science and Electrical Engineering
Henry H. Leitner, Senior Lecturer on Computer Science
Harry R. Lewis, Harvard College Professor and Gordon McKay Professor of Computer Science

Michael D. Mitzenmacher, Gordon McKay Professor of Computer Science *(on leave fall term)*
John G. Morrisett, Allen B. Cutting Professor of Computer Science
Radhika Nagpal, Assistant Professor of Computer Science on the Gordon McKay Endowment
Venkatesh Narayanamurti, John A. and Elizabeth S. Armstrong Professor of Engineering and Applied Sciences, Professor of Physics, Dean of the Division of Engineering and Applied Sciences and Dean for the Physical Sciences
Anthony G. Oettinger, Gordon McKay Professor of Applied Mathematics and Professor of Information Resources Policy
David C. Parkes, John L. Loeb Associate Professor of the Natural Sciences
Avrom J. Pfeffer, Associate Professor of Computer Science on the Gordon McKay Endowment
Michael O. Rabin, Thomas J. Watson, Sr. Professor of Computer Science
Norman Ramsey, Associate Professor of Computer Science on the Gordon McKay Endowment
Mema Roussopoulos, Assistant Professor of Computer Science on the Gordon McKay Endowment
Shimon Schocken, Visiting Professor of Computer Science *(The Interdisciplinary Center Herzliya) (fall term only)*
Margo I. Seltzer, Harvard College Professor and Herchel Smith Professor of Computer Science
Stuart M. Shieber, Harvard College Professor and James O. Welch, Jr. and Virginia B. Welch Professor of Computer Science
Michael D. Smith, Gordon McKay Professor of Computer Science and Electrical Engineering *(on leave spring term)*
Salil P. Vadhan, Thomas D. Cabot Associate Professor of Computer Science
Leslie G. Valiant, T. Jefferson Coolidge Professor of Computer Science and Applied Mathematics *(on leave 2005-06)*
James H. Waldo, Lecturer on Computer Science *(spring term only)*
Gu-Yeon Wei, Assistant Professor of Electrical Engineering on the Gordon McKay Endowment
Matthew D. Welsh, Assistant Professor of Computer Science on the Gordon McKay Endowment
Todd Zickler, Assistant Professor of Electrical Engineering on the Gordon McKay Endowment

The Division of Engineering and Applied Sciences (www.deas.harvard.edu) offers undergraduate and graduate courses in Applied Mathematics, Applied Physics, Computer Science, Earth and Planetary Sciences, and Engineering Sciences. Recommended course programs at the undergraduate level may be obtained from the Academic Office, Pierce Hall 110. Division faculty also offer several courses in the section entitled General Education Electives.

Primarily for Undergraduates

For information concerning concentration in Computer Science please consult the Director of Undergraduate Studies or the Academic Office, Division of Engineering and Applied Sciences, Pierce Hall 110. The Applied Mathematics and Engineering Sciences sections of the catalog should be consulted for additional courses relevant to computer science. In addition, attention is called to the following courses in related fields: Quantitative Reasoning 20; Applied Mathematics 106, 107; Linguistics 112a, 112b; Philosophy 144; Physics 123; and Statistics 110, 111, 171.

Computer Science 1. Great Ideas in Computer Science
Catalog Number: 6903

Henry H. Leitner
Half course (spring term). M., W., F., at 11. EXAM GROUP: 4
An introduction to the most important discoveries and intellectual paradigms in computer science, designed for students with little or no previous background. Explores problem-solving using high and low-level programming languages; presents an integrated view of computer systems, from switching circuits up through compilers and GUI design. Examines theoretical and practical limitations related to unsolvable and intractable computational problems, and the social and ethical dilemmas presented by such issues as software unreliability and invasions of privacy.
Note: May not be taken for credit after completing Computer Science 50.

Computer Science 50. Introduction to Computer Science I
Catalog Number: 4949
Michael D. Smith
Half course (fall term). M., W., F., at 10. EXAM GROUP: 3
Introduction to the intellectual enterprises of computer science. Algorithms: their design, specification, and analysis. Software development: problem decomposition, abstraction, data structures, implementation, debugging, testing. Architecture of computers: low-level data representation and instruction processing. Computer systems: programming languages, compilers, operating systems. Computers in the real world: networks, security and cryptography, artifical intelligence, social issues. Laboratory exercises include extensive programming in the C language.
Note: No previous computer experience required. This course, when taken for a letter grade, meets the Core area requirement for Quantitative Reasoning.

Computer Science 51. Introduction to Computer Science II
Catalog Number: 3411
Radhika Nagpal and Margo I. Seltzer
Half course (spring term). Tu., Th., 1–2:30. EXAM GROUP: 15, 16
Abstract models for computational processes and their concrete realizations. Functional, imperative, object-oriented and event-driven styles of programming. The structure, interpretation and compilation of programming languages. The engineering of complex software through procedural and data abstractions. Laboratory exercises using LISP (Scheme) and C++.
Prerequisite: Computer Science 50 or equivalent.

***Computer Science 91r. Supervised Reading and Research**
Catalog Number: 0361
Steven J. Gortler and David M. Brooks
Half course (fall term; repeated spring term). Hours to be arranged.
Supervised individual study of advanced topics in computer science. A student wishing to enroll in Computer Science 91r must be accepted by a faculty member who will supervise the course work. A form available from Pierce Hall 110, must be filled out and signed by the student and faculty supervisor. Students writing theses may enroll in this course while conducting thesis research and writing.
Note: At most two terms of Computer Science 91r may be taken for academic credit. May not be taken Pass/Fail. Students wishing more information about the range of suitable projects or faculty supervisors should consult the Director of Undergraduate Studies.

[*Computer Science 96. System Design Projects]
Catalog Number: 7499 Enrollment: Limited to 20.

Half course (fall term). Tu., Th., 2:30–4. EXAM GROUP: 16, 17
Cooperative design, development, and testing of a sizable and realistic computer network system. Students gain experience both in software development and system lifecycle issues, and in the area of application. We concentrate on mathematical modeling for prediction. The target application is prediction of student enrollments based on historical data, as raised by the recent discussions of preregistration. Students work in groups; both student participation in the classroom and effective group cooperation outside the classroom are stressed.
Note: Expected to be given in 2006–07.
Prerequisite: Computer Science 51.

For Undergraduates and Graduates

Computer Science 101. Digital Systems Construction
Catalog Number: 3166
Shimon Schocken (The Interdisciplinary Center Herzliya)
Half course (fall term). W., F., 3:30–5. EXAM GROUP: 8, 9
A computer science synthesis that starts with Nand gates and ends with applications. Students will build a general-purpose hardware platform using a simple hardware description language, then develop a compiler for a simple object-based language, and build a mini OS. Finally, they will play and develop interactive games like Pong and Tetris on the platform. All the necessary computer science knowledge is covered in the course; the only prerequisite is some programming experience.
Prerequisite: Computer Science 50 or equivalent.

Computer Science 121. Introduction to Formal Systems and Computation
Catalog Number: 0669
Salil P. Vadhan
Half course (fall term). Tu., Th., 10–11:30. EXAM GROUP: 12, 13
General introduction to formal systems and the theory of computation. Elementary treatment of automata, formal languages, computability, uncomputability, computational complexity, NP–completeness, and mathematical logic.

Computer Science 124. Data Structures and Algorithms
Catalog Number: 5207
Michael D. Mitzenmacher
Half course (spring term). Tu., Th., 11:30–1. EXAM GROUP: 13, 14
Design and analysis of efficient algorithms and data structures. Algorithm design methods, graph algorithms, approximation algorithms, and randomized algorithms are covered.
Prerequisite: Computer Science 50 or equivalent; Computer Science 51 is helpful. Some exposure to discrete applied mathematics, such as Applied Mathematics 106 or 107 or Computer Science 121 or Statistics 110, is also helpful.

Computer Science 141. Computing Hardware

Catalog Number: 4357
David M. Brooks
Half course (fall term). M., W., 1–2:30, and a two-hour weekly laboratory. EXAM GROUP: 6, 7
Introduction to the design, structure, and operation of digital computers; logic circuits and digital electronics; computer arithmetic; computer architecture; and machine language programming. Consideration of the design interactions between hardware and software systems.
Prerequisite: Programming experience required.

Computer Science 143. Computer Networks

Catalog Number: 6401
H. T. Kung
Half course (fall term). W., F., 1–2:30. EXAM GROUP: 6, 7
Principles, design, implementation, and performance of computer networks. Topics include: Internet protocols and routing, local area networks, TCP, performance analysis, congestion control, network address translation, voice and video over IP, switching and routing, mobile IP, peer-to-peer overlay networks, network security, and other current research topics. Programming assignments on protocol implementation and analysis.
Prerequisite: Computer Science 51.

Computer Science 144r. Networks Design Projects

Catalog Number: 5415
H. T. Kung
Half course (spring term). W., F., 2:30–4. EXAM GROUP: 7, 8
Cooperative design and development of advanced network-based systems with both technology and business considerations. Students will work in 2 person teams. Student work will include reading assignments, homework sets, a project proposal, and project reports and presentations. At the end of the class, all teams will defend their approaches and results in front of the class and invited guests.
Note: Preference given to upper-class undergraduates or graduate students in computer science or in business.
Prerequisite: Computer Science 143 or equivalent experience.

Computer Science 148. Design of VLSI Circuits and Systems

Catalog Number: 1772 Enrollment: Limited to 16.
Gu-Yeon Wei
Half course (spring term). Tu., Th., 11:30–1. EXAM GROUP: 13, 14
Presentation of concepts and techniques for the design and fabrication of VLSI systems and digital MOS integrated circuits. Topics include: basic semiconductor theory; MOS transistors and digital MOS circuits design; synchronous machines, clocking, and timing issues; high-level description and modeling of VLSI systems; synthesis and place and route design flows; and testing of VLSI circuits and systems. Various CAD tools for design, simulation, and verification are extensively used.
Prerequisite: Computer Science 141 or permission of instructor.

Computer Science 152. Principles of Programming Languages

Catalog Number: 6841
Norman Ramsey
Half course (spring term). M., W., F., at 11. EXAM GROUP: 4
Intellectual tools needed to design, evaluate, and choose programming languages. Historical influences on language design. Case studies, reinforced by programming exercises. Advanced languages, abstraction mechanisms. Includes functional, object-oriented, and logic paradigms. Focuses on practice, but covers formal topics crucial for intellectual rigor: abstract syntax, lambda calculus, type systems, and dynamic semantics. Grounding sufficient to read professional literature.
Prerequisite: Computer Science 121. Students must have excellent programming skills, be comfortable with recursion, basic mathematical ideas and notations.

Computer Science 153. Principles of Programming Language Compilation

Catalog Number: 2842
John G. Morrisett
Half course (fall term). M., W., F., at 11. EXAM GROUP: 4
Implementation of efficient interpreters and compilers for programming languages. Associated algorithms and pragmatic issues. Emphasizes practical applications including those outside of programming languages proper. Also shows relationships to programming-language theory and design. Participants build a working compiler including lexical analysis, parsing, type checking, code generation, and register allocation. Exposure to run-time issues and optimization.
Prerequisite: Computer Science 51.

Computer Science 161. Operating Systems

Catalog Number: 4347
Matthew D. Welsh
Half course (spring term). Tu., Th., 1–2:30. EXAM GROUP: 15, 16
The fundamental principles of resource management and abstraction in modern operating systems. Control abstractions: threads, processes, scheduling, synchronization. Storage abstractions: dynamic memory allocation, virtual memory, file system design. Communication abstractions: interprocess communication, networking. Case studies. Design and implementation of parts of a multiuser multitasking virtual-memory operating system.
Prerequisite: Computer Science 51.

Computer Science 164. Internet Technologies

Catalog Number: 7295
Mema Roussopoulos
Half course (fall term). M., W., 2:30–4. EXAM GROUP: 7, 8
Survey of current authoring, distributing, and browsing technologies used in the Internet. Topics include: HTTP, DNS and TCP/IP overview, HTML techniques for text, links, forms, and images, client/server paradigm, server-side programming, CGI scripts, dynamic content with Java, how web browsers and web servers work, web caching and replication.
Prerequisite: Computer Science 50.

[Computer Science 165. Information Management]
Catalog Number: 0560
Margo I. Seltzer
Half course (fall term). Tu., Th., 11:30–1. EXAM GROUP: 13, 14
Covers the fundamental concepts of database and information management. Data models: relational, object-oriented, and other; implementation techniques of database management systems, such as indexing structures, concurrency control, recovery, and query processing; management of unstructured data; terabyte-scale databases.
Note: Expected to be given in 2006–07.
Prerequisite: Computer Science 51.

Computer Science 175. Computer Graphics
Catalog Number: 3771
Steven J. Gortler
Half course (fall term). M., W., 4–5:30. EXAM GROUP: 9
The computational aspects of computer graphics. Two major themes are image rendering (viewing transformations, clipping, visible-surface processing, raster algorithms, reflection models, lighting models, surface shading, antialiasing, ray tracing, radiosity, and volume rendering) and scene modeling (modeling transformations, curves and surfaces, texture mapping, data-amplification techniques, constructive solid geometry, scalar- and vector-field data, and animation). Ancillary topics include color compression, image compression, image compositing, graphical user interfaces, and special machine architectures for computer graphics.
Prerequisite: Computer Science 51, Applied Mathematics 21b or Mathematics 21b.

Computer Science 181. Intelligent Machines: Perception, Learning, and Uncertainty
Catalog Number: 6454
Avrom J. Pfeffer
Half course (spring term). M., W., 2:30–4. EXAM GROUP: 7, 8
Introduction to artificial intelligence, focusing on problems of perception, machine learning and reasoning under uncertainty. Supervised learning algorithms. Neural networks and applications to character recognition. Statistical pattern recognition. Bayesian networks: representation, inference and learning. Hidden Markov models and applications to speech recognition. Markov decision processes and reinforcement learning.
Prerequisite: Computer Science 51 and Computer Science 121. Statistics 110 is recommended.

Computer Science 182. Intelligent Machines: Reasoning, Actions, and Plans
Catalog Number: 0134
David C. Parkes
Half course (fall term). M., W., 2:30–4. EXAM GROUP: 7, 8
Introduction to AI, focused on problems in reasoning about action and rational decision making. Search: constraint satisfaction; informed search and optimization; game playing. Knowledge representation and logical inference. Planning: representation, search and heuristics. Bounded rationality, situated agents. Multiagent systems. Discussion of relevant work in philosophy, economics, and decision theory. Applications to scheduling, robotics and e-commerce.
Prerequisite: Computer Science 51; Computer Science 121 (may be taken concurrently).

Computer Science 187. Computational Linguistics

Catalog Number: 0249

Stuart M. Shieber

Half course (fall term). Tu., Th., 10–11:30. EXAM GROUP: 12, 13

Introduction to computational linguistics, the study of human language using the tools and techniques of computer science, with applications to a variety of natural-language-processing problems. Representing syntactic structure: context-free, augmented context-free, and trans-context-free grammars. Representing semantic structure: first-order and higher-order logics. Computing with syntactic and semantic representations: Prolog programming; parsing and generation algorithms. Low-level language processing with finite-state methods.

Prerequisite: Computer Science 121.

Primarily for Graduates

Computer Science 220r. Cryptography: Trust and Adversity

Catalog Number: 1637

Michael O. Rabin

Half course (fall term). Tu., Th., 11:30–1. EXAM GROUP: 13, 14

Modern cryptography. Mathematical foundations. Public-key encryptions, digital signatures, key exchanges, zero-knowledge proofs, authentication, oblivious transfer, secure multi-party computations. Quantum and information theoretic secure encryptions. Foundations: Probablistic encryption and semantic security. Attacks and countermeasures.

Computer Science 221. Computational Complexity

Catalog Number: 5812

Salil P. Vadhan

Half course (spring term). Tu., Th., 2:30–4. EXAM GROUP: 16, 17

A quantitative theory of the resources needed for computing and the impediments to efficient computation. The models of computation considered include ones that are finite or infinite, deterministic, randomized, quantum or nondeterministic, discrete or algebraic, sequential or parallel.

Prerequisite: Computer Science 121 or equivalent.

[Computer Science 222. Algorithms at the Ends of the Wire]

Catalog Number: 2493

Michael D. Mitzenmacher

Half course (fall term). Tu., Th., 1–2:30. EXAM GROUP: 15, 16

Covers topics related to algorithms for big data, especially related to networks. Themes include compression, cryptography, coding, and information retrieval related to the World Wide Web. Requires a major final project.

Note: Expected to be given in 2006–07.

Prerequisite: Computer Science 124.

[Computer Science 223. Probabilistic Analysis and Algorithms]

Catalog Number: 4740

Michael D. Mitzenmacher

Half course (fall term). Tu., Th., 1–2:30. EXAM GROUP: 15, 16
Probabilistic techniques and tools for the design and analysis of algorithms. Designed for all first-year graduate students in all areas.
Note: Expected to be given in 2007–08.
Prerequisite: Computer Science 124. Preferably additional probability, such as in Computer Science 226r, Statistics 110, or Mathematics 191.

[Computer Science 224r. Randomness in Computation]
Catalog Number: 3380
Michael O. Rabin
Half course (fall term). Tu., Th., 11:30–1. EXAM GROUP: 13, 14
The surprising efficacy of randomization in the solution of algorithmic and computer science problems. Applications include number theory, cryptography, finite fields, computational geometry, routing, parallel algorithms, pattern matching, distributed systems, self-checking programs, probabilistically checkable proofs.
Note: Expected to be given in 2006–07.

[Computer Science 225. Pseudorandomness]
Catalog Number: 4869
Salil P. Vadhan
Half course (spring term). Tu., Th., 1–2:30. EXAM GROUP: 15, 16
Efficiently generating objects that “look random” despite being constructed using little or no randomness. Connections and applications to computational complexity, cryptography, and combinatorics. Pseudorandom generators, randomness extractors, expander graphs, error-correcting codes, hash functions.
Note: Expected to be given in 2006–07.
Prerequisite: Exposure to randomized algorithms (as in Computer Science 124), computational complexity (as in Computer Science 121), and algebra (as in Applied Mathematics 106, Mathematics 123, or Computer Science 226r).

[Computer Science 226r. Efficient Algorithms]
Catalog Number: 1749
Michael O. Rabin
Half course (fall term). Tu., Th., 11:30–1. EXAM GROUP: 13, 14
A survey of important computer algorithms for numerical and data manipulation problems and their applications in actual computing situations. Topics include combinatorial algorithms, string matching, wavelet algorithms, FFT and its applications, algebraic computations, randomized algorithms in algebra number theory and geometry, maximal flows, error correcting codes, public key cryptography, protocols for distributed systems, and parallel algorithms.
Note: Expected to be given in 2007–08.

[Computer Science 228. Computational Learning Theory]
Catalog Number: 0364
Leslie G. Valiant
Half course (spring term). Tu., Th., 2:30–4. EXAM GROUP: 16, 17
Possibilities of and limitations to performing learning by computational agents. Topics include

computational models, polynomial time learnability, learning from examples and learning from queries to oracles. Computational limitations. Statistical limitations. Applications to Boolean functions, automata and geometric functions. Learning algorithms for models of neural computation.
Note: Expected to be given in 2006–07.
Prerequisite: Computer Science 121 or equivalent.

Computer Science 229r (formerly Computer Science 229). Topics in the Theory of Computation
Catalog Number: 3730
Dan Gutfreund
Half course (spring term). M., W., 1–2:30. EXAM GROUP: 6, 7
Students read, present, and critically evaluate current research papers in theoretical computer science. See syllabus and web site for specific topics of focus.
Note: Expected to be omitted in 2006–07.
Prerequisite: Computer Science 121 or equivalent.

Computer Science 244r. Advanced Networks Design Projects
Catalog Number: 3018
H. T. Kung
Half course (spring term). W., F., 2:30–4. EXAM GROUP: 7, 8
The contents and course requirements are similar to those of Computer Science 144r, with the exception that students enrolled in Computer Science 244r are expected to do substantial system implementation and perform graduate-level work.
Note: Preference given to upper-class undergraduates or graduate students in computer science or in business who are proficient in computer programming or in business software.
Prerequisite: Computer Science 143 or equivalent experience.

Computer Science 246r. Advanced Computer Architecture
Catalog Number: 0979
David M. Brooks
Half course (spring term). M., W., 1–2:30. EXAM GROUP: 6, 7
Covers technology trends in computer system design, with an emphasis on power-aware computing for mobile, embedded, and traditional systems. System design areas include implementation, architecture, system software, and applications.
Note: Taught seminar style after the first several lectures.
Prerequisite: Computer Science 141 recommended. Consult instructor with questions.

[Computer Science 252r. Advanced Topics in Programming Languages]
Catalog Number: 1986
Norman Ramsey
Half course (fall term). W., F., 2:30–4. EXAM GROUP: 7, 8
Advanced functional programming. Lazy evaluation, monads. Folds and unfolds. Combinators for parsing and prettyprinting. Modules systems. Type systems: polymorphism and overloading, type and constructor classes, higher-order kinds, higher-rank polymorphism, polytypic programming. Implementation: heap profiling.

Note: Expected to be given in 2006–07.
Prerequisite: Computer Science 152 or permission of the instructor.

[Computer Science 253r. Advanced Topics in Programming Language Compilation]
Catalog Number: 2901 Enrollment: Limited to 18.
Michael D. Smith
Half course (spring term). Tu., Th., 10–11:30. EXAM GROUP: 12, 13
In-depth introduction to computer optimization. Topics include scalar optimization, register allocation, instruction scheduling, dependence analysis, interprocedural analysis, and cache optimization.
Note: Expected to be given in 2006–07. Preference given to graduate students or upper-class concentrators.
Prerequisite: Computer Science 153 or equivalent.

[Computer Science 255. Topics in Language-Based Security]
Catalog Number: 6216
John G. Morrisett
Half course (spring term). W., F., 2:30–4. EXAM GROUP: 7, 8
Reviews research in programming language-based security mechanisms. Topics include compiler and run-time techniques for enforcing policies; type and proof systems for expressing policies; and static analyses for establishing policies.
Note: Expected to be given in 2006–07. Taught in seminar style.
Prerequisite: Computer Science 152, 153, 252r, 253r, or permission of the instructor.

Computer Science 256. Programming Language and Semantics
Catalog Number: 1554
John G. Morrisett
Half course (spring term). W., F., 2:30–4. EXAM GROUP: 7, 8
An overview of operational, denotational, and axiomatic semantics; type systems, program analysis, and program equivalence.
Prerequisite: Computer Science 152.

Computer Science 257. Programming with Concurrency
Catalog Number: 9894
Norman Ramsey
Half course (fall term). W., F., 2:30–4. EXAM GROUP: 7, 8
Concurrency: theory, program structure, implementation. Emphasis to follow interests of participants. Threads, communicating processes, synchronization, transactions, concurrent functional programs. Modelling and checking. Implementation: synchronization, stack management, scheduling, garbage collection, heap-allocated activations, first-class continuations.
Prerequisite: Computer Science 152, 161, or 165, or permission of instructor.

Computer Science 261. Research Topics in Operating Systems
Catalog Number: 6706
Margo I. Seltzer
Half course (fall term). Tu., Th., 1–2:30. EXAM GROUP: 15, 16

A quantitative approach to operating system design and evaluation. Discussion of recent research including extensible operating system architectures, distributed systems, and performance analysis. Overview of research techniques and methodology.
Prerequisite: Computer Science 161, or equivalent.

Computer Science 262. Introduction to Distributed Computing
Catalog Number: 7949
James H. Waldo
Half course (spring term). M., W., 4–5:30. EXAM GROUP: 9
Examination of the special problems associated with distributive computing (e.g., partial failure and lack of global knowledge) and protocols that function in the face of these problems. Emphasis on causal ordering, event and RPC-based systems.
Prerequisite: Computer Science 161 or permission of instructor.

Computer Science 263. Wireless Sensor Networks
Catalog Number: 6846
Matthew D. Welsh
Half course (fall term). Tu., Th., 2:30–4. EXAM GROUP: 16, 17
Recent advances in wireless communications and sensor networks. Wireless networking, routing, standards including 802.11, Bluetooth, and 802.15.4. Embedded OS, programming tools, applications, and security. Students read research papers and undertake a research project.
Prerequisite: Computer Science 161 or Computer Science 143.

Computer Science 264. Peer-to-Peer Systems
Catalog Number: 6069 Enrollment: Limited to 24.
Mema Roussopoulos
Half course (spring term). Tu., Th., 11:30–1. EXAM GROUP: 13, 14
Discusses research papers on peer-to-peer systems. Topics include: routing, search, caching, security, reputation and trust, incentives, and applications. Students undertake a major research project and lead discussions of readings.
Note: Preference to graduate students or upper-level concentrators.
Prerequisite: Computer Science 161 or Computer Science 143.

[Computer Science 266. Biologically-Inspired Distributed and Multi-Agent Systems]
Catalog Number: 0766 Enrollment: Limited to 20.
Radhika Nagpal
Half course (fall term). Tu., Th., 10–11:30. EXAM GROUP: 12, 13
Surveys biologically-inspired approaches to designing distributed systems. Focus is on algorithms, analysis, and programming paradigms. Topics: swarm intelligence, amorphous computing, immune-inspired systems, synthetic biology. Discussion of research papers and a research project required.
Note: Expected to be given in 2006–07. Geared toward graduate students of all levels as well as advanced undergraduates. Preference given to graduate students or upper-level concentrators.
Prerequisite: Computer Science 161 or Computer Science 143 required.

Computer Science 277. Geometric Modeling in Computer Graphics
Catalog Number: 3067
Steven J. Gortler
Half course (spring term). M., W., 4–5:30. EXAM GROUP: 9
Advanced seminar in computer graphics focusing on geometric representations and processing. Topics include: direct manipulation, implicit surfaces, spline presentations, recursively subdivided surfaces, model simplification, surface parameterization and processing, mesh generation, and motion capture processing.
Prerequisite: Computer Science 175.

[Computer Science 278. Rendering and Image Processing in Computer Graphics]
Catalog Number: 4883

Half course (spring term). Tu., Th., 2:30–4. EXAM GROUP: 16, 17
Advanced course in computer graphics focusing on image rendering and processing. Topics include: light transport, efficient rendering, image based rendering, texture processing, interactive image processing.
Note: Expected to be given in 2006–07.
Prerequisite: Computer Science 175 or permission of instructor.

***Computer Science 279r. Topics in User Interfaces: Privacy and Security Usability**
Catalog Number: 1435 Enrollment: Limited to 12.
Stuart M. Shieber
Half course (spring term). Tu., Th., 10–11:30. EXAM GROUP: 12, 13
Seminar on topics drawn from computer-human interfaces, information retrieval, and information visualization. Intensive lab component emphasizes small group design and implementation. Spring 2006 focus is usability of computer security and privacy systems.
Prerequisite: Computer Science 51 and experience developing large software systems as evidenced by successful completion of a systems course requiring a large project.

[Computer Science 281r. Artificial Intelligence: Reasoning and Planning Systems]
Catalog Number: 0707
Avrom J. Pfeffer
Half course (fall term). M., W., 1–2:30. EXAM GROUP: 6, 7
In-depth introduction to formalisms for knowledge representation and techniques for reasoning and planning. Topics: formal logic-based representations; probabilistic reasoning; nonmonotonic logics; truth-maintenance systems; qualitative reasoning; inheritance hierarchies; computational approaches to reasoning about actions and time, including actions of multiple agents, nonlinear planning, plan recognition; reasoning about knowledge, belief, and action.
Note: Expected to be given in 2006–07.
Prerequisite: Computer Science 182 or permission of instructor.

Computer Science 282. Probabilistic Reasoning
Catalog Number: 3158
Avrom J. Pfeffer
Half course (fall term). Tu., Th., 1–2:30. EXAM GROUP: 15, 16

In-depth study of principles and techniques for probabilistic reasoning. Topics include: Bayesian networks and Markov networks; exact and approximate inference algorithms; learning Bayesian networks from data; temporal probability models; integrating logic and probability; influence diagrams.
Prerequisite: Computer Science 181 or permission of instructor.

Computer Science 283. Computer Vision

Catalog Number: 4475
Todd Zickler
Half course (fall term). M., W., F., at 10. EXAM GROUP: 3
Vision as an ill-posed inverse problem: image formation, two-dimensional signal processing; image enhancement and restoration; feature analysis; image segmentation; structure from motion, texture, and shading; multiple view geometry; pattern classification; and applications.

[Computer Science 285. Multi-agent Planning Systems]

Catalog Number: 1060
Barbara J. Grosz
Half course (spring term). Tu., Th., 1–2:30. EXAM GROUP: 15, 16
Theories and techniques for multi-agent planning, including formal models of rational agents, collaborative plans, and social systems; computational approaches to distributed planning and problem solving, negotiation, and decision theory for planning; collaborative systems design.
Note: Expected to be given in 2006–07.
Prerequisite: Computer Science 181 or 182, or permission of instructor.

Computer Science 286r. Topics at the Interface between Computer Science and Economics

Catalog Number: 1099 Enrollment: Limited to 20.
David C. Parkes
Half course (spring term). M., W., 1–2:30. EXAM GROUP: 6, 7
Interplay between computation and incentives within open decentralized computational systems. Mechanisms and market design, negotiation, social-choice, information-economics and privacy. Readings from theoretical CS, AI, operations research, and economics. Seminar style. Spring 2006: Multi-agent Learning and Implementation.
Note: Preference given to graduate students or upper-class concentrators.
Prerequisite: Mathematics 21b, Applied Mathematics 21b, or equivalent; Computer Science 121, 124, and 181 or 182, or equivalents; or permission of instructor.

[Computer Science 287r. Natural Language Processing]

Catalog Number: 3306
Stuart M. Shieber
Half course (fall term). Tu., Th., 2:30–4. EXAM GROUP: 16, 17
In-depth investigation of natural-language-processing techniques. Topics include: finite-state, context-free, and trans-context-free formalisms, syntactic analysis, semantic interpretation, weighted automata and transducers. Students discuss research papers and undertake a significant research project.
Note: Expected to be given in 2006–07.
Prerequisite: Computer Science 187 or permission of instructor.

Computer Science 288. Computational Models of Discourse

Catalog Number: 1392
Barbara J. Grosz
Half course (spring term). Tu., Th., 1–2:30. EXAM GROUP: 15, 16
Computational theories of discourse structure and processing. Topics include: anaphora, focusing, speech acts, collaborative planning and plan recognition algorithms, intonation. Application to dialogue and text-processing systems and design of human-computer interface systems.
Prerequisite: Computer Science 182, 187, or 287r or equivalent, or permission of instructor.

Computer Science 299r. Special Topics in Computer Science

Catalog Number: 4592
Michael D. Smith (fall term) and Margo I. Seltzer (spring term)
Half course (fall term; repeated spring term). Hours to be arranged.
Supervision of experimental or theoretical research on acceptable computer science problems and supervision of reading on topics not covered by regular courses of instruction.
Note: Open to graduate students and AB/SM candidates only. Students must arrange such work with a member of the Division. This course is graded and is ordinarily taken with the approval of the Committee on Higher Degrees. Applicants must file a project sheet before study cards are filed. Project sheets may be obtained from the Academic Office, Pierce Hall 110.

Graduate Courses of Reading and Research

Reading courses are odd-numbered; research courses are even-numbered.

***Computer Science 305,306. Information Resources: Technology and Policy**
Catalog Number: 6364,3478
Anthony G. Oettinger 2403

***Computer Science 307,308. Biologically-Inspired Multi-Agent Systems, Distributed Systems, and Computational Biology**
Catalog Number: 8289,8308
Radhika Nagpal 5068

***Computer Science 309,310. Computational Mechanism Design, Electronic Marketplaces, and Multi-Agent Systems**
Catalog Number: 8764,0931
David C. Parkes 4202

***Computer Science 311,312. Collaborative Systems, AI Planning, and Natural Language Processing**
Catalog Number: 4677,6223
Barbara J. Grosz 1599

***Computer Science 317,318. Distributed Systems, Networking, and Mobile Computing**
Catalog Number: 9388,7137
Mema Roussopoulos 4822

***Computer Science 319,320. Distributed Systems, Operating Systems, and Networks**
Catalog Number: 8038,8568
Matthew D. Welsh 4600

***Computer Science 321,322. Databases, Operating System, and Software Design**
Catalog Number: 4085,4086
Margo I. Seltzer 3371

***Computer Science 323,324. Human-Computer Communication through Natural, Graphical, and Artificial Languages**
Catalog Number: 2450,2453
Stuart M. Shieber 2456

***Computer Science 325,326. Programming Languages and Tools**
Catalog Number: 8055,0747
Norman Ramsey 2831

***Computer Science 327,328. Mathematical Logic, Theory of Computation**
Catalog Number: 1160,3576
Harry R. Lewis 4455

***Computer Science 343,344. Computer Architecture: Modeling and Design**
Catalog Number: 3932,9266
David M. Brooks 4222

***Computer Science 345,346. High-Performance Computer Systems**
Catalog Number: 6154,6156
Michael D. Smith 3372 (on leave spring term)

***Computer Science 347,348. Computer Vision**
Catalog Number: 1882,8831
Todd Zickler 5143

***Computer Science 351,352. Complexity of Computations: Concurrent Programming and Synchronization**
Catalog Number: 0218,0255
Michael O. Rabin 7003

***Computer Science 353,354. Representation and Reasoning, Machine Learning and Decision Making**
Catalog Number: 6816,1843
Avrom J. Pfeffer 2830