7575 (on leave spring term), Francisco Márquez 5064, Sandra Naddaff 7779, Gregory Nagy 1423 (on leave spring term), Stephen Owen 7418, Judith Ryan 1135, Marc Shell 3176 (on leave fall term), Susan R. Suleiman 7234 (on leave 2000-01), Maria Tatar 3645 (on leave 2000-01), William Mills Todd III 1634 (on leave spring term), and Ruth R. Wisse 3177

***Comparative Literature 397. Direction of Doctoral Dissertations**
Catalog Number: 0320
Jan Ziolkowski 7275, Margaret Alexiou 1214, Sacvan Bercovitch 7638 (on leave fall term), Svetlana Boym 1926, Joaquim-Francisco Coelho 7715 (on leave spring term), James Engell 8076, Luis M. Girón Negrón 3060, George G. Grabowicz 4511, Karl S. Guthke 1715, Barbara E. Johnson 7626 (on leave fall term), Walter Kaiser 2561, Robert Kiely 1621, James L. Kugel 7575 (on leave spring term), Francisco Márquez 5064, Sandra Naddaff 7779, Gregory Nagy 1423 (on leave spring term), Stephen Owen 7418, Judith Ryan 1135, Marc Shell 3176 (on leave fall term), Susan R. Suleiman 7234 (on leave 2000-01), Maria Tatar 3645 (on leave 2000-01), William Mills Todd III 1634 (on leave spring term), and Ruth R. Wisse 3177

***Comparative Literature 399. Reading and Research**
Catalog Number: 2893
Jan Ziolkowski 7275, Margaret Alexiou 1214, Sacvan Bercovitch 7638 (on leave fall term), Svetlana Boym 1926, Joaquim-Francisco Coelho 7715 (on leave spring term), James Engell 8076, Luis M. Girón Negrón 3060, George G. Grabowicz 4511, Karl S. Guthke 1715, Barbara E. Johnson 7626 (on leave fall term), Walter Kaiser 2561, Robert Kiely 1621, James L. Kugel 7575 (on leave spring term), Francisco Márquez 5064, Sandra Naddaff 7779, Gregory Nagy 1423 (on leave spring term), Stephen Owen 7418, Judith Ryan 1135, Marc Shell 3176 (on leave fall term), Susan R. Suleiman 7234 (on leave 2000-01), Maria Tatar 3645 (on leave 2000-01), William Mills Todd III 1634 (on leave spring term), and Ruth R. Wisse 3177
Note: Candidates for the doctoral degree in Comparative Literature may pursue advanced studies under the individual supervision of these instructors. Permission to register for this course should be obtained from the instructor whose guidance is sought and from the Chairman of the Department.

Computer Science

AN HISTORICAL EDITION OF FAS COURSES OF INSTRUCTION

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

Michael S. Brandstein, Associate Professor of Electrical Engineering on the Gordon McKay Endowment
James L. Frankel, Lecturer on Computer Science
Steven J. Gortler, Associate Professor of Computer Science on the Gordon McKay Endowment *(Director of Undergraduate Studies)*
Barbara J. Grosz, Higgins Professor of Natural Sciences
H. T. Kung, William H. Gates Professor of Computer Science and Electrical Engineering
Henry H. Leitner, Senior Lecturer on Computer Science
Harry R. Lewis, Gordon McKay Professor of Computer Science and Dean of Harvard College
Michael D. Mitzenmacher, Assistant Professor of Computer Science on the Gordon McKay Endowment
Venkatesh Narayanamurti, John A. and Elizabeth S. Armstrong Professor of Engineering and Applied Sciences and Professor of Physics *(Dean of the Division of Engineering and Applied Sciences)*
Anthony G. Oettinger, Gordon McKay Professor of Applied Mathematics and Professor of Information Resources Policy
Avrom J. Pfeffer, Assistant Professor of Computer Science on the Gordon McKay Endowment
Michael O. Rabin, Thomas J. Watson, Sr. Professor of Computer Science
Norman Ramsey, Assistant Professor of Computer Science on the Gordon McKay Endowment
Margo I. Seltzer, Gordon McKay Professor of Computer Science
Stuart M. Shieber, Gordon McKay Professor of Computer Science *(on leave 2001-2002)*
Michael D. Smith, Gordon McKay Professor of Computer Science and Electrical Engineering *(on leave spring term)*
S. Tucker Taft, Visiting Lecturer on Computer Sciences
Salil P. Vadhan, Assistant Professor of Computer Science on the Gordon McKay Endowment
Leslie G. Valiant, T. Jefferson Coolidge Professor of Computer Science and Applied Mathematics
James H. Waldo, Lecturer on Computer Science
Woodward Yang, Gordon McKay Professor of Electrical Engineering and Computer Science *( )*

The Division of Engineering and Applied Sciences offers undergraduate and graduate courses in Applied Mathematics, Applied Physics, Computer Science, Earth and Planetary Sciences, and Engineering Sciences. Recommended curricula may be obtained from the Academic Office, Pierce Hall 110a. 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 110a. 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: General Education 156, Linguistics 112a, 112b, Philosophy 144, Physics 123, Statistics 110, 111, 171.

Computer Science 50. Introduction to Computer Science I
Catalog Number: 4949

Stuart M. Shieber
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 and experimenting with and analyzing software systems.
Note: No previous computer experience required.

Computer Science 51. Introduction to Computer Science II
Catalog Number: 3411
Henry H. Leitner
Half course (spring term). Tu., Th., 1–2:30. EXAM GROUP: 15, 16
Abstract models for computational processes and their concrete realizations. Functional, imperative, and object-oriented styles of programming; processor and memory architectures; interpretation and compilation of programming languages. State-space search, finite-state processes, formal logic, data and functional abstraction, and syntactic and semantic formalisms as examples of useful abstractions. The engineering of complex software. Laboratory exercises using LISP, C++, and Java.
Prerequisite: Computer Science 50 or equivalent.

***Computer Science 91r. Supervised Reading and Research**
Catalog Number: 0361
Steven J. Gortler
Half course (fall term; repeated spring term). Hours to be arranged.
In this course a student may undertake supervised individual study of advanced topics in computer science beyond those covered in regular courses, or may participate in a computer science research project. Students writing theses may enroll in this course while conducting their thesis research and writing. A student wishing to enroll in Computer Science 91r must be accepted by a faculty member who will supervise the course work and will specify the syllabus or project description. A form available in the Division of Engineering and Applied Sciences Academic Office, Pierce Hall 110a, must be filled out with a description of the course work and the basis for its evaluation. This form must be signed by the student and the faculty supervisor and filed in the Academic Office by the date on which study cards are due. A written report of the work carried out in the course is ordinarily required by the beginning of the reading period.
Note: Ordinarily, 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.

For Undergraduates and Graduates

Computer Science 121. Introduction to Formal Systems and Computation
Catalog Number: 0669
Harry R. Lewis

Half course (fall term). Tu., Th., 10–11:30. EXAM GROUP: 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). M., W., 1–2:30. EXAM GROUP: 6, 7
Design and analysis of efficient algorithms. Data structure representations and their use for provably efficient implementation of abstract operations: searching, sorting, set manipulation. Memory management. Graph algorithms. General algorithm design techniques.
Prerequisite: Computer Science 51; some exposure to discrete applied mathematics, such as Applied Mathematics 106 or 107 or Computer Science 121 or Statistics 110, is helpful.

Computer Science 141. Computing Hardware
Catalog Number: 4357
Michael D. Smith
Half course (fall term). M., W., F., at 9, and laboratory hours to be arranged. EXAM GROUP: 2
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: Computer Science 50.

Computer Science 143. Computer Networks
Catalog Number: 6401
H. T. Kung
Half course (fall term). M., W., 2:30–4. EXAM GROUP: 7, 8
Architecture, design, and performance of computer networks. Topics include: the Internet protocols, local area networks, performance analysis, queueing theory, congestion control, multicast, quality of service, and network security. Programming exercises on protocol implementation.
Prerequisite: Computer Science 51.

Computer Science 144r (formerly Computer Science 144). Networks Design Projects
Catalog Number: 5415
H. T. Kung
Half course (spring term). W., F., 4–5:30. EXAM GROUP: 9
Cooperative design and development of a computer network or network-based service based on new and promising concepts which may still be under research. Exploration of real-world design concerns, including survey and critiques of relevant literature, early validation of proposed approach, design specification, implementation, testing, and evaluation. Students work in groups, and present weekly status reports. At the end of the class, students will defend their approaches and results in the presence of experts.
Note: Enrollment is Limited. Preference given to concentrators in Computer Science who are

proficient in computer programming.
Prerequisite: Computer Science 143 or equivalent experience.

Computer Science 146. Computer Architecture
Catalog Number: 6520
James L. Frankel
Half course (spring term). Tu., 2:30–5:30. EXAM GROUP: 16, 17, 18
Review of the fundamental structures in modern processor design. Topics include computer organization, instruction set design, memory system design, pipelining, and other techniques to exploit parallelism. Emphasis on a quantitative evaluation of design alternatives and an understanding of timing issues.
Prerequisite: Computer Science 141.

Computer Science 148. Introduction to VLSI Design
Catalog Number: 1772 Enrollment: Limited to 16.
Edward J. Burdick
Half course (spring term). M., 2:30–5:30. EXAM GROUP: 7, 8, 9
Presentation of concepts and techniques for the design and fabrication of VLSI integrated circuits. Topics include: basic semiconductor theory; pn junctions; MOS transistors; integrated circuit fabrication technology; VLSI layout; digital MOS circuit design; memory and processor design; and testing of VLSI circuits. CAD tools for design and simulation are extensively used for homework assignments and for a large project assignment. High quality projects may be fabricated at an external VLSI foundry.
Prerequisite: Engineering Sciences 50 or Physics 15b, and Computer Science 141, or permission of instructor.

Computer Science 152. Principles of Programming Languages
Catalog Number: 6841
Norman Ramsey
Half course (fall term). M., W., F., at 11. EXAM GROUP: 4
Intellectual tools needed to design, evaluate, and choose programming languages. Historical influence of theory, software engineering, and implementation technique on language design. Case studies, reinforced by programming exercises. Emphasizes advanced languages, abstraction mechanisms. Includes functional, object-oriented, and logic paradigms. Focuses on ideas and techniques most relevant to practitioners, but covers theoretical topics crucial for intellectual rigor: specification based on 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. Must be comfortable with recursion and with basic mathematical ideas and notations.

Computer Science 153. Principles of Programming Language Compilation
Catalog Number: 2842
S. Tucker Taft
Half course (spring term). Tu., Th., 4:30–6. EXAM GROUP: 18
The underlying theory of the implementation of interpreters and compilers for programming languages, associated algorithms, and pragmatic issues. Theoretical emphasis on the relation to

programming language theory and practical emphasis on applications outside of programming language implementation proper. Topics include lexical analysis, parsing algorithms, type checking and inference, code generation, run-time issues, optimization.
Prerequisite: Computer Science 121 and 152.

Computer Science 161. Operating Systems
Catalog Number: 4347
Margo I. Seltzer
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: thread, 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.
Note: Open to students who achieved an honor grade (B- or better) in Computer Science 51 and who have experience developing large software systems.
Prerequisite: Computer Science 51.

[Computer Science 165. Introduction to Database Systems]
Catalog Number: 4712

Half course (fall term). Hours to be arranged.
Design principles for modern distributed database systems. Topics include: extended E/R, relational and object-oriented data models; query processing, persistence, concurrency control, back-up and recovery; database connectivity; Java and XML languages; Web information organization, indexing and retrieval; search engines architecture and algorithms.
Note: Expected to be given in 2001–02.
Prerequisite: Computer Science 161 or permission of instructor.

Computer Science 175. Computer Graphics
Catalog Number: 3771
Steven J. Gortler
Half course (fall term). W., F., 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, Computer Science 121 and Statistics 110, or equivalent.

Computer Science 182. Intelligent Machines: Reasoning, Actions, and Plans
Catalog Number: 0134
Barbara J. Grosz
Half course (fall term). M., W., 1–2:30. EXAM GROUP: 6, 7
Introduction to AI focused on approaches to problems of reasoning about action. Search and game-playing. Knowledge representation. Partial-order planning: representations of actions; techniques for handling goal interactions. Resource-limited planning; situated agents. Discussion of relevant work in philosophy and decision theory; applications to vision, language, robotics.
Prerequisite: Computer Science 51; Computer Science 121 (may be taken concurrently).

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
Topics in modern cryptography. Primality testing, finite fields, elliptic curves. Protocols: Public-key encryptions, digital signatures, key exchanges, zero-knowledge proofs, authentication oblivious transfer, secret sharing, proactive security, fair contract signing, distributed agreements. Foundations: Probablistic encryption and semantic security. Attacks and countermeasures: Non-malleabilty, plaintext awareness and proofs of plaintext knowledge. Absolutely secure encryptions. Prerequisites will be discussed in sections.

[Computer Science 221. Computational Complexity]
Catalog Number: 5812
Leslie G. Valiant
Half course (spring term). Hours to be arranged.
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, probabilistic, quantum or nondeterministic, discrete or algebraic, sequential or parallel.
Note: Expected to be given in 2001–02.
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). Hours to be arranged.
Covers topics related to what is done with information before and after it is sent across a

network. Themes include compression, cryptography, coding, and information retrieval related to the World Wide Web. Theoretical aspects are emphasized, although current practice and recent advances are also a focus. Requires a major final project.
Note: Expected to be given in 2001–02.
Prerequisite: Computer Science 124.

Computer Science 223. Probabilistic Analysis and Algorithms
Catalog Number: 4740
Michael D. Mitzenmacher
Half course (fall term). Tu., Th., 2:30–4. EXAM GROUP: 16, 17
The course will focus on how Markov chains and random processes are used to analyze algorithms and network behavior. Reading current research in the area will be required. Topics may include heavy-tailed distributions, load balancing, stochastic bin-packing, and models of the Web.
Prerequisite: Computer Science 124. Preferably additional probability, such as in Computer Science 224r, Computer Science 226r, Statistics 110, or Mathematics 191.

[Computer Science 224r. Randomness in Computation]
Catalog Number: 3380
Michael O. Rabin
Half course (fall term). Hours to be arranged.
Exploration of the surprising efficacy of randomization in the solution of algorithmic and general computer science problems. Applications include number theoretic algorithms, cryptographic protocols, computations in finite fields, computational geometry. CS applications will include routing in networks, parallel algorithms, pattern matching, agreement protocols for distributed systems. We shall also deal with programs that check and correct their own work and with Probabilistically Checkable Proofs (PCP). The probability theory prerequisites will be covered.
Note: Expected to be given in 2001–02.

[Computer Science 226r. Efficient Algorithms]
Catalog Number: 1749
Michael O. Rabin
Half course (fall term). Hours to be arranged.
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, 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 2002–03.

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.
Prerequisite: Computer Science 121 or equivalent.

Computer Science 244r (formerly Computer Science 244). Advanced Networks Design Projects
Catalog Number: 3018
H. T. Kung
Half course (spring term). W., F., 4–5:30. EXAM GROUP: 9
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 devise novel algorithms and protocols, and give research presentations. In addition, substantial implementation and documentation are required.
Note: Enrollment is limited. Preference given to graduate students, or upper-class concentrators, in Computer Science who are proficient in computer programming.
Prerequisite: Computer Science 143 or equivalent experience.

[*Computer Science 246 (formerly Computer Science 246r). Advanced Computer Architecture]
Catalog Number: 0979

Half course (spring term). Hours to be arranged.
The contents and course requirements are similar to those of Computer Science 146, with the exception that students enrolled in Computer Science 246 are required to conduct extra readings and to complete an additional term project.
Note: Expected to be given in 2001–02.
Prerequisite: Background in computer software and hardware, and permission of the instructor.

Computer Science 252r. Advanced Topics in Programming Languages
Catalog Number: 1986
Norman Ramsey
Half course (spring term). Tu., Th., 10–11:30. EXAM GROUP: 12, 13
Advanced functional programming. Lazy evaluation, monads, monad comprehensions, the monadic approach to imperative features. Folds and unfolds. Functional reactive programming for graphics, robotics. Combinators for parsing and prettyprinting. Purely functional data structures. Type systems: polymorphism and overloading, type and constructor classes, higher-order kinds, polytypic programming. Implementation: heap profiling, match compilation.
Prerequisite: Computer Science 152 or permission of the instructor.

[Computer Science 253. Advanced Principles of Programming Language Compilation]
Catalog Number: 2901
Michael D. Smith
Half course (spring term). Hours to be arranged.
In-depth introduction to compiler optimizations developed to exploit recent advances in computer architecture. Topics include scalar optimization, instruction scheduling for superscalar

and VLIW processors, data dependence analysis, interprocedural analysis on both array and pointer variables, cache optimizations such as blocking and prefetching.
Note: Expected to be given in 2001–02.
Prerequisite: Computer Science 153 or equivalent.

Computer Science 254r. Programming Methodologies
Catalog Number: 2767
Robert L. Walton
Half course (spring term). W., 4–6. EXAM GROUP: 9
Investigates program analysis, verification, and refinement; programming paradigms, including parallel and distributed; program development and maintenance environments. This year students will critique an experimental world-wide programming environment the instructors are developing: see www.deas.harvard.edu/courses/cs254r/2001.
Prerequisite: Computer Science 51 and 121, or equivalent.

[Computer Science 261. Research Topics in Operating Systems]
Catalog Number: 6706
Margo I. Seltzer
Half course (fall term). .
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.
Note: Expected to be given in 2001–02.
Prerequisite: Computer Science 161, or equivalent.

[Computer Science 262. Introduction to Distributed Computing]
Catalog Number: 7949
James H. Waldo
Half course (spring term). Hours to be arranged.
Examination of the special problems associated with distributive computing, especially those associated with partial failure and intrinsic limitations on global knowledge. The course will emphasize the specification and implementation of high level protocols that allow computational entities to collaborate in the face of these problems. Causal ordering, event and RPC based systems, and security problems in distributed systems will be discussed.
Note: Expected to be given in 2001–02.
Prerequisite: Computer Science 161 or permission of instructor.

Computer Science 265r (formerly Computer Science 265). Database Systems
Catalog Number: 4104
Margo I. Seltzer
Half course (fall term). Tu., Th., 1–2:30. EXAM GROUP: 15, 16
A research-oriented introduction to Database Management systems. First third covers database design, implementation, and use. Topics include: network, relational, and object oriented database models, system architectures, transaction processing, system implementation, and SQL. Remaining two-thirds address research literature surrounding database systems, including an historical perspective, the emergenceof relational and object-oriented systems, concurrency

control, and distributed systems. Students will be expected to undertake a final research project.
Prerequisite: Computer Science 51.

Computer Science 275. Advanced Computer Graphics
Catalog Number: 5495
Steven J. Gortler
Half course (fall term). W., F., 4–5:30. EXAM GROUP: 9
The contents and course requirements are similar to those of Computer Science 175, with the exception that students enrolled in Computer Science 275 are required to solve more difficult problem sets.
Prerequisite: Computer Science 51, Applied Mathematics 21b or Mathematics 21b.

Computer Science 276r. Computer Graphics, Special Topics
Catalog Number: 8097
Steven J. Gortler
Half course (spring term). M., W., 2:30–4. EXAM GROUP: 7, 8
Seminar examining in detail some specific aspect of computer graphics. Specific topics which change from year to year may include: image based rendering, photo-realistic rendering, geometric rerpresentations, representations of motion and animations, computer graphics hardware. Students will make one oral presentation, and create a software implementation of one of the covered concepts.
Prerequisite: Computer Science 175 or 275, or permission of instructor.

[Computer Science 279. Topics in Computer-Human Interfaces, Information Retrieval and Visualization]
Catalog Number: 2407 Enrollment: Enrollment may be limited.
Stuart M. Shieber
Half course (spring term). Hours to be arranged.
Seminar providing background and current research in specific topics drawn from one or more of computer-human interfaces, information, retrieval, and information visualization. Intensive lab component emphasizes small group design and implementation of systems in these areas.
Note: Expected to be given in 2001–02.
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). Hours to be arranged.
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 2001–02.
Prerequisite: Computer Science 182, or permission of instructor.

Computer Science 282. Probabilistic Reasoning
Catalog Number: 3158
Avrom J. Pfeffer
Half course (fall term). M., W., 2:30–4. EXAM GROUP: 7, 8
In-depth study of principles and techniques for probabilistic reasoning and decision-theoretic planning. Topics include: Bayesian networks and Markov networks; exact and approximate probabilistic inference algorithms; learning Bayesian networks from data; temporal probability models; integrating logic and probability; influence diagrams; Markov decision processes; reinforcement learning.
Prerequisite: Computer Science 181 or permission of instructor.

[Computer Science 283. Computer Vision]
Catalog Number: 4475
Michael S. Brandstein
Half course (fall term). Hours to be arranged.
Vision as an ill-posed inverse problem: two-dimensional signal processing; image enhancement and restoration; feature analysis; image segmentation and analysis; structure from motion, texture, and shading; binocular stereo; pattern classification; and applications.
Note: Expected to be given in 2001–02.

Computer Science 285. Multi-agent Planning Systems
Catalog Number: 1060
Barbara J. Grosz
Half course (spring term). Tu., Th., 11:30–1. EXAM GROUP: 13, 14
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.
Prerequisite: Computer Science 182 or permission of instructor.

***Computer Science 287r. Natural Language Processing**
Catalog Number: 3306
Stuart M. Shieber
Half course (spring term). Tu., Th., 2:30–4. EXAM GROUP: 16, 17
Principles and techniques of natural language processing, including grammar formalisms, syntactic analysis, semantic interpretation, and associated algorithms.
Prerequisite: Computer Science 121 and 152.

[Computer Science 288. Computational Models of Discourse]
Catalog Number: 1392
Barbara J. Grosz
Half course (spring term). Hours to be arranged.
Computational theories of discourse (text and dialogue) structure and processing. Topics include: anaphora, focusing, plans and speech acts, plan recognition algorithms, models of collaborative planning, intonation. Discussion of dialogue and text understanding systems. Application to the design of human-computer interface systems.

Note: Expected to be given in 2001–02.
Prerequisite: Computer Science 182 or 287r or equivalent, or permission of instructor.

Computer Science 299r. Special Topics in Computer Science
Catalog Number: 4592
Venkatesh Narayanamurti
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 A.B./S.M. candidates only. Students must arrange such work with a member of the Division. This course is ordinarily taken with the approval of the Committee on Higher Degrees in certain cases when a letter grade is required. Applicants should file a project sheet before study cards are filed. Project sheets may be obtained from the Academic Office, Pierce Hall 110a.

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

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

***Computer Science 323,324. Programming Languages, Natural Language Processing, and Human-Computer Interfaces**
Catalog Number: 2450,2453
Stuart M. Shieber 2456 (on leave 2001-2002)

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

***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

***Computer Science 355,356. Computational Complexity, Parallel Computation, Machine Learning, and Neural Computation**
Catalog Number: 0345,0346
Leslie G. Valiant 7396

***Computer Science 357. Computational Complexity, Cryptography, and Pseudorandomness**
Catalog Number: 3485
Salil P. Vadhan 3833 (spring term only)

***Computer Science 358. Computational Complexity, Cryptography, and Pseudorandomness**
Catalog Number: 8641
Salil P. Vadhan 3833 (spring term only)

***Computer Science 359,360. Online Algorithms and Randomized Algorithms**
Catalog Number: 2104,1477
Michael D. Mitzenmacher 7748

***Computer Science 375,376. Computer Graphics**
Catalog Number: 6832,7313
Steven J. Gortler 2824

Courses Related to Ethnic Studies