



[Comparative Literature 266 (formerly Literature 149). Irony]
[Comparative Literature 288. Antiquity and Beyond: Modern Critical Theory and the Classics - \(New Course\)](#)

Comparative Literature

AN HISTORICAL EDITION OF FAS COURSES OF INSTRUCTION

The Department of Comparative Literature has merged with the Committee on Degrees in Literature to form the Department of Literature and Comparative Literature. Please see the Department of Literature and Comparative Literature.

Computer Science

AN HISTORICAL EDITION OF FAS COURSES OF INSTRUCTION

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

David M. Brooks, John L. Loeb Associate Professor of the Natural Sciences

Sevan G. Ficici, Lecturer on Computer Science

Yakov Gal, Lecturer on Computer Science

Steven J. Gortler, Robert I. Goldman Professor of Computer Science (*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, Harvard College Professor and Gordon McKay Professor of Computer Science (*on leave spring term*)

David J. Malan, Lecturer on Computer Science

Michael D. Mitzenmacher, Gordon McKay Professor of Computer Science , Contin Ed/Spec Prog Instructor

John G. Morrisett, Allen B. Cutting Professor of Computer Science, Associate Dean for



Computer Science and Engineering

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 School of Engineering and Applied Sciences
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
Hanspeter Pfister, Gordon McKay Professor of the Practice of Computer Science, Director of Visual Computing in the Initiative in Innovative Computing
Michael O. Rabin, Thomas J. Watson, Sr. Professor of Computer Science
Margo I. Seltzer, Harvard College Professor and Herchel Smith Professor of Computer Science
Stuart M. Shieber, James O. Welch, Jr. and Virginia B. Welch Professor of Computer Science
Michael D. Smith, Gordon McKay Professor of Computer Science and Electrical Engineering, Dean of the Faculty of Arts and Sciences
Lynn A. Stein, Visiting Professor of Computer Science (*Olin College*)
Salil P. Vadhan, Gordon McKay Professor of Computer Science and Applied Mathematics (*on leave 2007-08*)
Leslie G. Valiant, T. Jefferson Coolidge Professor of Computer Science and Applied Mathematics
James H. Waldo, Lecturer on Computer Science
Gu-Yeon Wei, Associate Professor of Electrical Engineering on the Gordon McKay Endowment
Matthew D. Welsh, Associate Professor of Computer Science on the Gordon McKay Endowment
Todd Zickler, Assistant Professor of Electrical Engineering on the Gordon McKay Endowment

The School of Engineering and Applied Sciences (www.seas.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. Engineering and Applied Sciences faculty also offer several courses in the section entitled Freshman Seminars, Extra-Departmental Courses, and House Seminars.

Primarily for Undergraduates

For information concerning concentration in Computer Science please consult the Director of Undergraduate Studies or the Academic Office, School 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. May be used for Quantitative Reasoning Core credit.

Computer Science 50. Introduction to Computer Science I

Catalog Number: 4949

David J. Malan

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, artificial intelligence, social issues. Assignments include extensive programming in the C language and PHP.

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

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

Abstraction and design in computation. Topics include: Functional and object-oriented styles of programming; software engineering in the small; implementation of a language interpreter. Goal: understanding how to design large programs to make them readable, maintainable, efficient, and elegant. Exercises in LISP (Scheme) and C++.

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.

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 the Academic Office, 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.



For Undergraduates and Graduates

Computer Science 61 (formerly Computer Science 160). Systems Programming and Machine Organization - (New Course)

Catalog Number: 3461

Matthew D. Welsh

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

Fundamentals of computer systems programming, machine organization, and performance tuning. This course provides a solid background in systems programming and a deep understanding of low-level machine organization and design. Topics include C and assembly language programming, program optimization, memory hierarchy and caching, virtual memory and dynamic memory management, concurrency, threads, and synchronization.

Prerequisite: Computer Science 50

[Computer Science 120. Introduction to Cryptography]

Catalog Number: 5911

Salil P. Vadhan

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

Algorithms to guarantee privacy and authenticity of data during communication and computation. Rigorous proofs of security based on precise definitions and assumptions. Topics may include one-way functions, private-key and public-key encryption, digital signatures, pseudorandom generators, higher-level protocols such as electronic cash, and the role of cryptography in network and systems security.

Note: Expected to be given in 2008–09.

Prerequisite: Computer Science 121 or Computer Science 124.

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: 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 (spring 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 (fall term). M., W., 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. Programming Languages

Catalog Number: 6841

John G. Morrisett



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.

Note: Expected to be given in 2008–09.

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.

Note: Expected to be given in 2008–09.

Prerequisite: Computer Science 50.

Computer Science 165. Information Management

Catalog Number: 0560



Margo I. Seltzer

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

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.

Prerequisite: Computer Science 51.

Computer Science 171. Visualization - (New Course)

Catalog Number: 8877

Hanspeter Pfister

Half course (spring term). M., W., 1–2:30 plus weekly section to be arranged. EXAM GROUP: 6, 7

Introduction to key design principles and techniques for visualizing data. Covers data and image models, visual perception, interaction techniques, animation, tools from various fields, and design practices. Introduces programming of static, dynamic, and interactive visualizations.

Prerequisite: Computer Science 50 or equivalent, Mathematics 1b. Exceptions by permission of the instructor.

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 179. Design of Usable Interactive Systems - (New Course)**

Catalog Number: 4052 Enrollment: Limited to 24.

Lynn A. Stein (Olin College)

Half course (spring term). M., W., 2:30–4:30. EXAM GROUP: 7, 8, 9

Usability and design as keys to successful technology. Covers user observation techniques, needs assessment, low and high fidelity prototyping, usability testing methods, as well as design best practices. Focuses on understanding and applying the lessons of human interaction to the design of usable systems; will also look at lessons to be learned from less usable systems. The course centers on a semester-long design project, with classes mixing studio and seminar formats.

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). M., W., 1–2:30. EXAM GROUP: 6, 7

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.

[Computer Science 199r. Special Topics in Computer Science]

Catalog Number: 4242

Michael D. Smith

Half course (fall term; repeated spring term). Tu., Th., 2:30-4. EXAM GROUP: Fall: 16, 17

Topic focus for 2007: Privacy and Technology. Case studies of areas in which there are perceived conflicts between individual privacy and computer technology. Which of these conflicts are real? Which could reasonably be addressed through changes in policy and technology? Areas include RFID, surveillance, biometrics, data aggregation and data mining. Open to all students.

Note: Expected to be given in 2008–09.

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 tools. Public-key encryptions, digital signatures, key exchanges, zero-knowledge proofs, authentication, oblivious transfer, electronic elections, auctions, secure multi-party computations. Quantum and provably secure encryptions. Foundations: Probabilistic encryption and semantic security. Attacks and countermeasures.

Computer Science 221. Computational Complexity

Catalog Number: 5812

Leslie G. Valiant

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 2008–09.

Prerequisite: Computer Science 124.

Computer Science 223. Probabilistic Analysis and Algorithms

Catalog Number: 4740

Michael D. Mitzenmacher

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

Probabilistic techniques and tools for the design and analysis of algorithms. Designed for all first-year graduate students in all areas.

Prerequisite: Computer Science 124. Preferably additional probability, such as in Computer Science 226r, Statistics 110, or Mathematics 191.

[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 2008–09.

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

Important algorithms and their real life applications. Topics include combinatorics, string matching, wavelets, FFT, computational algebra number theory and geometry, randomized algorithms, search engines, maximal flows, error correcting codes, cryptography, distributed and parallel algorithms.

Note: Expected to be given in 2008–09.

[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. Applications to Boolean functions, automata and geometric functions.

Note: Expected to be given in 2008–09.

Prerequisite: Computer Science 121 or equivalent.

[Computer Science 229r. Topics in the Theory of Computation]

Catalog Number: 3730

Salil P. Vadhan

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 given in 2008–09.

Prerequisite: Computer Science 121 or equivalent.

Computer Science 244r. Networks Design Projects

Catalog Number: 3018

H. T. Kung

Half course (fall term). M., W., 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). Th., at 1, Tu., 5:30–7:30 p.m. EXAM GROUP: 15, 18

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 248. Advanced Design of VLSI Circuits and Systems**

Catalog Number: 7191 Enrollment: Limited to 16.

Gu-Yeon Wei

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

The contents and course requirements are similar to those of Computer Science 148, with the exception that students enrolled in Computer Science 248 are expected to do a substantial design project and paper discussions on advanced topics.

Prerequisite: Computer Science 141 or permission of instructor.

[Computer Science 250r. Topics in Programming Language Design and Implementation]

Catalog Number: 8553

John G. Morrisett

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

Seminar course discussing readings from research in programming language design and implementation. This offering will explore unifying abstractions for next-generation programming languages. Transactions and communication, types and effects, types and logics, modules and classes.

Note: Expected to be given in 2008–09.

Prerequisite: Computer Science 152, Computer Science 153, or equivalent.

[Computer Science 260 (formerly Computer Science 260r). Topics in Computer Systems]

Catalog Number: 7764

Matthew D. Welsh

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

Readings from research literature in operating systems, distributed systems, and networking. The topic in 2006 will be "Internet-Scale Sensor Networking." Large-scale querying on Internet data; stream-based database systems; interfacing to sensor networks.

Note: Expected to be given in 2008–09.

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 263r (formerly 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.

Note: Expected to be given in 2008–09.

Prerequisite: Computer Science 161 or Computer Science 143.

[Computer Science 264. Peer-to-Peer Systems]

Catalog Number: 6069 Enrollment: Limited to 24.

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: Expected to be given in 2008–09. 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 16.

Radhika Nagpal

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

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: Geared toward graduate students of all levels as well as advanced undergraduates.

Preference given to graduate students or upper-level concentrators.

Prerequisite: Experience with algorithms (e.g. Computer Science 124) and programming (e.g. Computer Science 51).

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

Steven J. Gortler

Half course (spring term). M., W., 4–5:30. EXAM GROUP: 9

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 2008–09.

Prerequisite: Computer Science 175 or permission of instructor.

[*Computer Science 279 (formerly *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 2008 focus is usability of computer security and privacy systems.

Note: Expected to be given in 2008–09.

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 study of artificial intelligence techniques for reasoning, planning, and learning. Topics vary from year to year.

Note: Expected to be given in 2008–09.

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

Yakov Gal and Sevan G. Ficici

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.

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). F., 2–5. EXAM GROUP: 7, 8, 9

Spring 2008: Computational Finance. AI models of financial markets. Automated trading agents. Terabyte-scale financial data challenges. Derivative pricing; price prediction. Regulatory considerations. Readings in CS theory, AI, systems; operations research; finance. Applied final project.

Note: Preference given to graduate students or upper-class concentrators in computer science or economics.

Prerequisite: Mathematics 21b, Applied Mathematics 21b, or equivalent; and either a) Computer Science 124, and 181 or 182, b) Economics 1723 and 1760, or equivalents; or c) permission of instructor. A background in finance is helpful but not required.

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

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.

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.

Note: Expected to be given in 2008–09.

Prerequisite: Computer Science 182, 187, or 287r or equivalent, or permission of instructor.

Computer Science 299r. Special Topics in Computer Science

Catalog Number: 4592

John G. Morrisett

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 School of Engineering and Applied Sciences. 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 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 313,314. Visual Computing**

Catalog Number: 4273,1628

Hanspeter Pfister 5882



***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 327,328. Mathematical Logic, Theory of Computation**

Catalog Number: 1160,3576

Harry R. Lewis 4455 (on leave spring term)

***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

***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

***Computer Science 355,356. Computational Complexity, Parallel Computation, Computational Learning, Neural Computation**

Catalog Number: 0345,0346

Leslie G. Valiant 7396

***Computer Science 357,358. Computational Complexity, Cryptography, and Pseudorandomness**



Catalog Number: 3485,8641

Salil P. Vadhan 3833 (on leave 2007-08)

***Computer Science 359,360. On-line Algorithms and Randomized Algorithms**

Catalog Number: 2104,1477

Michael D. Mitzenmacher 7748

***Computer Science 361,362. Programming Languages and Semantics**

Catalog Number: 8672,8366

John G. Morrisett 4853

***Computer Science 375,376. Computer Graphics**

Catalog Number: 6832,7313

Steven J. Gortler 2824

Dramatic Arts

AN HISTORICAL EDITION OF FAS COURSES OF INSTRUCTION

Faculty of the Committee on Dramatics

Robert Scanlan, Professor of the Practice of Theater (*Chair*)

Elizabeth Weil Bergmann, Lecturer on Dramatic Arts

Deborah D. Foster, Senior Lecturer on Folklore and Mythology

Marjorie Garber, William R. Kenan, Jr. Professor of English and of Visual and Environmental Studies (*on leave spring term*)

Jorie Graham, Boylston Professor of Rhetoric and Oratory

Gideon M. Lester, Lecturer on Dramatic Arts

Elizabeth D. Lyman, Assistant Professor of English

Christie McDonald, Smith Professor of French Language and Literature and Professor of Comparative Literature

John C. Megan, Director, Office for the Arts at Harvard (*ex officio*)

Robert J. Orchard, Director of the Loeb Drama Center and Executive Director of the American Repertory Theatre (*ex officio*)

Peter Sacks, John P. Marquand Professor of English

Elaine Scarry, Walter M. Cabot Professor of Aesthetics and the General Theory of Value (*on leave spring term*)

Diana Sorensen, James F. Rothenberg Professor of Romance Languages and Literatures and of Comparative Literature, Dean for the Humanities