Department of Computer Science

Professor: *Hildreth[A], Shull (Chair)*

Associate Professor: *Metaxas, Turbak*

Assistant Professor: *Tjaden, Shaer*

Hess Visiting Assistant Professor: *Gold*

Lecturer: *Anderson*

Laboratory Instructor: *Herbst, Kakavouli, Lee*

Computer Science encompasses the systematic study of computing systems and computation. It is continually evolving and expanding, making it an exciting field of study. All of the traditional areas in CS as well as newer directions are represented in our faculty's expertise (including algorithms, programming languages, data structures, artificial intelligence, databases, computer architecture, networks, security, vision, graphics, parallel computing, robotics, bioinformatics, web information retrieval, multimedia), allowing us to offer a large variety of courses and substantial research opportunities for students.

The Computer Science Department offers three introductory computer science courses: CS 110, 111, and 112. For advice in making a choice consult "Choosing an Introductory CS Course" online at http://cs.wellesley.edu/~cs/Curriculum/whichCS1xx.html.

CS 110 Computer Science and the Internet
Anderson, Shull, Turbak, Staff
This course explains the basics of how the Internet works and how to build a Web site. Topics include client-server architecture, structuring Web pages with HTML, CSS, and JavaScript, the representation of colors, images, and sound on the computer, encryption, cookies, and CGI forms. We also discuss accessibility, copyright, intellectual property, and critical thinking in the context of the Internet. The required project models most phases of the standard software lifecycle. Students are introduced to programming by building an interactive Web site using JavaScript. Students are required to attend an additional discussion section each week. *Students considering additional computer science courses should take 111, not 110. Students can receive Mathematical Modeling distribution credit for only one of 110, 111, and 112.*

Prerequisite: None. No prior background with computers is expected.
Distribution: Mathematical Modeling
Semester: Fall, Spring Unit: 1.0

CS 111 Computer Programming and Problem Solving
Metaxas, Staff, Tjaden
An introduction to problem solving through computer programming. Using the Java programming language, students learn how to read, modify, design, debug, and test algorithms that solve problems. Programming concepts include control structures, data structures, abstraction, recursion, modularity, and object-oriented design. Students explore these concepts in the context of interactive programs involving graphics and user interfaces. *Students are required to attend an additional two-hour laboratory section each week. Required for students who wish to major or minor in computer science or elect more advanced courses in the field. Students can receive Mathematical Modeling distribution credit for only one of 110, 111, and 112.*

Prerequisite: None. No prior background with computers is expected.
Distribution: Mathematical Modeling. Does not satisfy the laboratory requirement.
Semester: Fall, Spring Unit: 1.0

CS 112 Computation for the Sciences
Tjaden
An introduction to computer programming that provides the tools necessary for students to use computers effectively in scientific work, physical sciences, biological sciences, medicine, mathematics, psychology and economics. Students learn to write software to solve problems, visualize and analyze data, perform computer simulations, and implement and test computational models that arise in a wide range of scientific disciplines. The course introduces MATLAB, an extensive and widely used technical computing environment with advanced graphics, visualization and analysis tools, and a rich high-level programming language. *Students are required to attend an additional two-hour laboratory section each week. Students can receive Mathematical Modeling distribution credit for only one of 110, 111, and 112.*

Prerequisite: None. No prior background with computers is expected.
Distribution: Mathematical Modeling
Semester: Spring Unit: 1.0

CS 115/PHYS 115 Robotic Design Studio (Wintersession)
Turbak, Berg (Physics)
In this intensive course, students are introduced to engineering principles while designing and assembling robots out of LEGO® parts, sensors, motors, and tiny computers. Fundamental robotics skills are learned in the context of studying and modifying a simple robot known as SciBorg. Then, working in small teams, students design and build their own robots for display at a robot exhibition. These projects tie together aspects of a surprisingly wide range of disciplines, including computer science, physics, math, biology, psychology, engineering, and art. *Students may register for either CS 115 or PHYS 115 and credit will be granted accordingly. Not offered every year. Subject to Dean's Office approval.*

Prerequisite: None
Distribution: Natural and Physical Science
Semester: Wintersession Unit: 0.5

As of 5/1/2008

CS 215 Multimedia Design and Programming
Metaxas
The purpose of this course is to give students a broad foundation in issues related to creating multimedia and hypermedia applications. Topics to be covered include history and philosophy of hypermedia; principles of human-computer interaction; multimedia programming; optimizing for CD-ROMs and the World Wide Web; digital representation and editing of media (audio, graphics, video); media compression and transmission; and delivery of multimedia applications.

Prerequisite: At least 111 (preferred) or 110 is required. At least one of ARTS 105, ARTS 108 or ARTS 109 is recommended.
Distribution: Mathematical Modeling
Semester: Fall, Spring Unit: 1.0

CS 230 Data Structures
Metaxas, Tjaden
An introduction to techniques and building blocks for organizing large programs. Topics include: modules, abstract data types, recursion, algorithmic efficiency, and the use and implementation of standard data structures and algorithms such as lists, trees, graphs, stacks, queues, priority queues, tables, sorting, and searching. Students become familiar with these concepts through weekly programming assignments using the Java programming language.

Prerequisite: 111 or permission of the instructor. Students who received a grade of C+ or lower in 111 must contact the instructor before enrolling.
Distribution: Mathematical Modeling
Semester: Fall, Spring Unit: 1.0

CS 231 Fundamental Algorithms
Shull
An introduction to the design and analysis of fundamental algorithms. General techniques covered: divide-and-conquer algorithms, dynamic programming, greediness, probabilistic algorithms. Topics include: sorting, searching, graph algorithms, compression, cryptography, computational geometry, and NP-completeness.

Prerequisite: 230 and either MATH 225 or permission of the instructor.
Distribution: Mathematical Modeling
Semester: Spring Unit: 1.0

CS 232 Artificial Intelligence
Gold
An introduction to artificial intelligence (AI), the design of computer systems that possess and acquire knowledge and can reason with that knowledge. Topics include knowledge representation, problem solving and search, planning, vision, language comprehension and production, learning, common sense reasoning, and expert systems. To attain a realistic and concrete understanding of these problems, CommonLisp, an AI programming language, will be taught and used to implement the algorithms of the course. *Alternate year course.*

Prerequisite: 230 or by permission of the instructor.
Distribution: Mathematical Modeling
Semester: Fall Unit: 1.0

CS 235 Languages and Automata
Turbak
An introduction to the concepts of languages and automata. Topics include languages, regular expressions, finite automata, grammars, pushdown automata, and Turing machines. The first half of the semester covers the Chomsky hierarchy of languages and their associated computational models. The second half of the semester focuses on decidability issues and unsolvable problems and the course closes with a brief introduction to complexity theory. The course includes a programming component investigating the application of automata theory to the scanning and parsing of programming languages.

Prerequisite: 230 and either MATH 225 or permission of the instructor.
Distribution: Mathematical Modeling
Semester: Fall Unit: 1.0

CS 240 Introduction to Machine Organization with Laboratory
Shull
Most students' experience with computers is limited to the highest level of the computer. This course is intended to demystify the computer (open up the "black box") and teach how information at the highest level is processed and ultimately executed by the underlying circuitry. To this end, the course provides an introduction to machine organization and assembly language programming. Specific topics include the fundamentals of computer organization (introduction to numeric representation, boolean logic, digital logic and all associated technology), a basic data path implementation, assembly language programming, how to assess and understand the performance of a computer, and brief overviews of assemblers, compilers and operating systems. *Students are required to attend one three-hour laboratory weekly.*

Prerequisite: 111
Distribution: Mathematical Modeling. This course satisfies the laboratory requirement.
Semester: Fall Unit: 1.25

CS 242 Computer Networks
Staff
NOT OFFERED IN 2008-09. OFFERED IN 2009-10. A systems-oriented approach to data networks, including a theoretical discussion of common networking problems and an examination of modern networks and protocols. Topics include point-to-point links, packet switching, internetworking, end-to-end protocols, congestion control, and security. Projects may include client-server applications and network measurement tools. *Alternate year course.*

Prerequisite: 230 or permission of the instructor.
Distribution: Mathematical Modeling
Semester: N/O. Offered in 2009-10. Unit: 1.0

CS 249 Topics in Computer Science
Shaer
Topic for 2008-09: Human-Computer Interaction. Human-Computer Interaction is one of the areas that have transformed the way we use computers in the last 30 years. Topics include methodology for designing and testing user interfaces, interaction styles (command line, menus, graphical user interfaces, virtual reality, tangible user interfaces), interaction techniques (including use of voice, gesture, eye movements), design guidelines, and user interface software tools. Students will design a user interface, program a prototype, and test the result for usability.

Prerequisite: Computer Science 110 or 111.
Distribution: Mathematical Modeling
Semester: Spring Unit: 1.0

CS 250 Research or Individual Study

Prerequisite: 230 or permission of the instructor.
Distribution: None
Semester: Fall, Spring Unit: 1.0

CS 250H Research or Individual Study

Prerequisite: 230 or permission of the instructor.
Distribution: None
Semester: Fall, Spring Unit: 0.5

CS 251 Theory of Programming Languages
Shull
An introduction to the dimensions of modern programming languages. Covers major programming paradigms: function-oriented, imperative, object-oriented, and logic-oriented. Dimensions include syntax, naming, state, data, control, concurrency, nondeterminism, and types. These dimensions are explored via mini-language interpreters written in OCaml, Scheme, and Haskell that students experiment with and extend.

Prerequisite: 230 and either 235 or permission of the instructor.
Distribution: Mathematical Modeling
Semester: Spring Unit: 1.0

CS 303/BISC 303 Bioinformatics
Tjaden
NOT OFFERED IN 2008-09. OFFERED IN 2009-10. A multidisciplinary course exploring the origins, present and future applications, and challenges of the intersection of biological and computer sciences. The field of bioinformatics, generated in response to the era of genomics, encompasses all aspects of biological data acquisition, storage, processing, analysis and interpretation with a view to generating in silico models of cellular function. *Students may register for either CS 303 or BISC 303 and credit will be granted accordingly.*

Prerequisites: 231, BISC 219 or BISC 220
Distribution: Natural and Physical Science
Semester: N/O. Offered in 2009-10. Unit: 1.0

CS 304 Databases with Web Interfaces
Anderson
A study of the 3-layer architecture commonly used for web-based applications such as e-commerce sites. We will learn to model and design databases uses entity-relationship diagrams, and the Standard Query Language (SQL) for managing databases. We will learn PHP, CGI/Perl, and Java Servlets, which are three important technologies for web-based architectures. We will also discuss performance, reliability and security issues. Finally, we will create dynamic Web sites driven by database entries. *Alternate year course.*

Prerequisite: 230
Distribution: Mathematical Modeling
Semester: Spring Unit: 1.0

CS 307 Computer Graphics
Anderson
NOT OFFERED IN 2008-09. OFFERED IN 2009-10. A survey of topics in computer graphics with an emphasis on fundamental techniques. Topics include: graphics hardware, fundamentals of three-dimensional graphics including modeling, projection, coordinate transformation, synthetic camera specification, color, lighting, shading, hidden surface removal, animation, and texture-mapping. We also cover the mathematical representation and programming specification of lines, planes, curves, and surfaces.

Prerequisite: 230
Distribution: Mathematical Modeling
Semester: N/O. Offered in 2009-10. Unit: 1.0

CS 310 Theoretical Foundations of Cryptology
Shull
NOT OFFERED IN 2008-09. OFFERED IN 2009-10. When is a cryptographic system secure and how will we ever know? This course introduces the computational models and theory computer scientists use to address these issues. Topics include one-way functions, trapdoor functions, probabilistic complexity classes, pseudorandom generators, interactive proof systems, zero-knowledge proofs, and the application of these theories to modern cryptology.

Prerequisite: 231 or 235 or permission of the instructor.
Distribution: Mathematical Modeling
Semester: N/O. Offered in 2009-10. Unit: 1.0

CS 332 Visual Processing by Computer and Biological Vision Systems
Hildreth
NOT OFFERED IN 2008-09. OFFERED IN 2009-10. An introduction to algorithms for deriving symbolic information about the three-dimensional environment from visual images. Aspects of models for computer vision systems will be related to perceptual and physiological observations on biological vision systems. Computer vision software written in MATLAB will be used to implement and test models. Topics include: edge detection, stereopsis, motion analysis, shape from shading, color, visual reasoning, object recognition. *Alternate year course.*

Prerequisite: 112 or 230 or permission of the instructor.
Distribution: Mathematical Modeling
Semester: N/O. Offered in 2009-10. Unit: 1.0

CS 342 Computer Security
Turbak
An introduction to computer security. Topics include ethics, privacy, authentication, access control, information flow, operating system security (with a focus on Linux), cryptography, security protocols, intrusion prevention and detection, firewalls, viruses, network security, Web security, programming language security. Assignments include hands-on exercises with security exploits and tools in a Linux environment. Participants will independently research, present, and lead discussions on security-related topics. *Alternate year course.*

Prerequisite: 242 or permission of the instructor. Not open to students who have taken this topic as CS 349.
Distribution: Mathematical Modeling
Semester: Fall Unit: 1.0

CS 343 Advanced Computer Organization and Design with Laboratory
Staff
NOT OFFERED IN 2008-09. OFFERED IN 2009-10. This course explores advanced computer organization and design. The interaction of hardware and software at a variety of levels is studied to provide a framework for how to design a computer. These ideas are also applied to understand how a computer system works and why it performs as it does. Examples and measurements based on commercial systems are used to create realistic design experiences. In addition, RISC architectures for current desktop, server, and embedded computers will also be surveyed. *Integral to the course is a three-hour required weekly laboratory*. In the laboratory, students write behavioral models of computer components using VHDL, a commercial computer design language. Most significantly, students will engage in an intensive, semester-long project in which they design a simple 16-bit RISC-style microprocessor. This includes the design of an instruction interpreter, a register file, a singlecycle, and a multicycle CPU. The lectures complement the labs. *Alternate year course.*

Prerequisite: 240. Not open to students who have taken this course as a topic of CS 349.
Distribution: Mathematical Modeling. This course satisfies the laboratory requirement.
Semester: N/O. Offered in 2009-10. Unit: 1.25

CS 349 Advanced Topics in Computer Science
Metaxas
Topic for 2008-09: Web Search and Mining. Locating information on the web seems very easy, while determining the quality of information can be tricky. This course is for students who want to know why search engines can answer your queries fast and (mostly) accurately, why other times they seem to be missing the point and provide untrustworthy information, and how one can design a website that acquires high visibility on the web. We will cover traditional information retrieval methods and web search algorithms such as crawlers, with a focus on methods that can detect web spam. We will also cover some basic understanding of text mining and data clustering. We will examine other relevant issues of the information explosion era, such as the shape and structure of the web, epistemology of information.
Prerequisite: 230
Distribution: Mathematical Modeling
Semester: Fall Unit: 1.0

CS 350 Research or Individual Study

Prerequisite: Open by permission to juniors and seniors.
Distribution: None
Semester: Fall, Spring Unit: 1.0

CS 350H Research or Individual Study

Prerequisite: Open by permission to juniors and seniors.
Distribution: None
Semester: Fall, Spring Unit: 0.5

CS 360 Senior Thesis Research

Prerequisite: By permission of the department. See Academic Distinctions.
Distribution: None
Semester: Fall, Spring Unit: 1.0

CS 370 Senior Thesis

Prerequisite: 360 and permission of department.
Distribution: None
Semester: Fall, Spring Unit: 1.0

Directions for Election

Students majoring in computer science must complete CS 111, 230, 231, 235, 240, 251, two 300-level courses other than 350, 360 or 370, and at least one additional computer science course at the 200 or 300 level. Students who do not take CS 111 must replace this requirement with one additional one-unit computer science course except 110. Computer science courses at other institutions used to meet the nine-course requirement must be approved **in advance** by the department chair on an individual basis. In addition, all majors in computer science are expected to complete (1) MATH 225 (Combinatorics and Graph Theory) and (2) at least one additional course in mathematics at the 200 or 300 level. Particularly relevant mathematics courses are MATH 206 (Linear Algebra), MATH 220 (Probability and Elementary Statistics), MATH 223 (Number Theory), MATH 305 (Abstract Algebra), MATH 309 (Foundations of Mathematics), and MATH 349 (Graph Theory). Students should consult a CS faculty member for advice in choosing mathematics courses best suited to their interests. Students are encouraged to consult the current computer science student handbook (linked from the department's web site http://cs.wellesley.edu) for suggestions of possible course schedules for completing the major. Students considering a junior year abroad should consult a faculty member in the department as soon as possible in their sophomore year to plan a schedule of courses to complete the major.

All computer science majors are required to participate in computer science student **seminars** held throughout the academic year. In this seminar, students have the opportunity to explore topics of interest through reading and discussion, field trips, invited speakers, independent research projects, or software development projects.

The computer science five-course minimum minor is recommended for students whose primary interests lie elsewhere, but who wish to obtain a fundamental understanding of computer science. The minor consists of CS 111, 230, either 231 or 235 or 240, at least one CS course above 100-level, and at least one 300-level CS course other than 350. Students who do not take CS 111 must replace this requirement with one additional one-unit computer science course except 110.

Students may receive a maximum of one unit of college credit for a score of 5 on the Computer Science A or AB Advanced Placement exam. This unit does not count towards the computer science major or minor. Students receiving AP credit for computer science should consult with the department regarding enrollment in 230 or 240. CS majors and minors should consult with a CS faculty advisor before electing to take a CS course as credit/non.

Students can earn **honors** in computer science by successfully completing an honors-quality senior thesis. A detailed description of the senior thesis project in computer science can be found in the document *Independent Studies in Computer Science* (http://cs.wellesley.edu/~cs/Research/independ.html). Majors who are interested in undertaking a senior thesis project are urged to discuss their plans with either their advisor or the department chair as early as possible in their junior year.

Computer Engineering
Students interested in computer engineering should consult the course listings in Extradepartmental and enroll in EXTD 160, Introduction to Engineering Science. This course is intended to be a gateway experience for possible subsequent engineering studies such as the engineering certificates from the Olin College of Engineering. The Special Academic Programs section contains a description of these certificates that represent groups of engineering courses at Olin designed to complement a major at Wellesley. More information at http://cs.welleseley.edu/~cs/Curriculum/olin.html.

Students who plan to pursue graduate work in computer science are strongly encouraged to develop their background in mathematics, particularly in the areas of linear algebra, probability and statistics, and graph theory. Such students should elect one or more of 310, 349 or MATH 305. In addition, students who are planning either graduate work or advanced technical research or development work are strongly encouraged to (1) obtain laboratory experience by electing one or more of 303, 307, 332, 342, 343, or appropriate courses at MIT and (2) pursue at least one independent study or research project before graduating, in the form of a Wellesley course (250/350/360), an MIT UROP, or a summer internship; consult http://cs.wellesley.edu/~cs/Research for more details.

Students interested in an **interdepartmental** major (or minor, if applicable) in cognitive and linguistic sciences, media arts and sciences, or neuroscience are referred to these listings in the catalog.

Students interested in engineering should consult the course listings in Extradepartmental.