A student may apply at the start of their senior year by submitting an application to the Computer Engineering M.S. program, provided that at the end of junior year, the student was a UCR Computer Engineering B.S. student with cumulative GPA at least 3.4 and had completed the following courses with no grade less than a B- and average grade at least 3.2: CS 100, CS 120A, CS 120B, CS 161, CS 161L. The application to the M.S. program must include at least two recommendation letters from UCR Academic Senate faculty members (at least one, and preferably both, CSE faculty). Submission of GRE scores with the application is recommended but not required. Matriculation into the combined program occurs in the Fall term following senior year, provided: (a) the M.S. application is accepted, (b) throughout senior year, the student is a Computer Engineering B.S. major with cumulative GPA 3.4 or higher, (c) by the end of senior year, the student completes the Computer Engineering B.S. degree requirements.

Incoming students who are applying to the Computer Engineering B.S. program may simultaneously apply for preliminary admission into the combined program provided their high school GPA is at least 3.6, their SAT-I combined score is at least 1950, they satisfy the Entry Level Writing requirement before matriculation, and they have sufficient math preparation to enroll in calculus upon arrival. Preliminary admission status is maintained as long as the student is a Computer Engineering or Computer Science B.S. student in good standing with a cumulative GPA of at least 3.4. Preliminarily admitted students still need to apply for full admission in their senior year as described above.

Five-year programs leading to M.S. degrees in other programs (including Computer Science) are also available. They are described separately in the catalog sections for those programs.

### Computer Science and Engineering

**Subject abbreviation:** CS  
**The Marian and Rosemary Bourns College of Engineering**

Laxmi N. Bhuyan, Ph.D., Chair  
Department Office,  
351 Winston Chung Hall  
(951) 827-5639; \[www.cs.ucr.edu\]

#### Distinguished Professor  
Laxmi N. Bhuyan, Ph.D.

**Professors**

- Marek Chrobak, Ph.D.
- Rajiv Gupta, Ph.D.
- Tao Jiang, Ph.D., President’s Chair
- Eamonn Keogh, Ph.D.
- Srikanth Krishnamurthy, Ph.D.
- Stefano Lonardi, Ph.D.
- Mart L. Møllec, Ph.D.
- Walid Najjar, Ph.D.
- Michael Pazzani, Ph.D.
- Chinja Ravishankar, Ph.D.
- Vassilis Isaros, Ph.D.
- Frank N. Vahid, Ph.D.
- Neal Young, Ph.D.

**Professors Emeriti**

- Yang-Chiang Hong, Ph.D.
- Lawrence L. Lamoree, Ph.D.
- Thomas H. Payne, Ph.D.
- Teodor C. Przymusinski, Ph.D.

**Associate Professors**

- Evangelos Christidis, Ph.D.
- Christian Shelton, Ph.D.
- Victor Zordan, Ph.D.

**Assistant Professors**

- Philip Brisk, Ph.D.
- Harsha Madhyastha, Ph.D.
- Iulian Neamtiu, Ph.D.
- Tamar Shinar, Ph.D.
- Zhihong Chen, Ph.D.

**Adjunct Professor**

Michalals Faloutsos, Ph.D.

**Cooperating Faculty**

- Bir Bhanu, Ph.D. (Electrical Engineering)
- Ilya Dumer, Ph.D. (Electrical Engineering)
- Thomas Girk, Ph.D. (Botany and Plant Sciences)
- Roger Lake, Ph.D. (Electrical Engineering)
- Michel L. Lapidus, Ph.D. (Mathematics)
- Erik Rolland, Ph.D. (Accounting and Information Systems)
- Amit Roy-Chowdury, Ph.D. (Electrical Engineering)
- Thomas Stahovich, Ph.D. (Mechanical Engineering)
- Sheldon Tan, Ph.D. (Electrical Engineering)
- Qi Zhu, Ph.D. (Electrical Engineering)

#### Major

The Department of Computer Science and Engineering offers three majors at the undergraduate level. UCR’s offerings of all three majors are unique compared to many schools in the emphasis on theoretical foundations and practical applications.

The **Computer Science** major stresses the study of core and advanced computer science topics. It prepares students for a large variety of careers in computing, including software engineering, networks, databases, graphics, algorithms, security, system analysis, and embedded systems.

The objective of the B.S. degree program in Computer Science is to prepare graduates for professional practice in both the private and public sectors and for life-long learning, including the option for graduate degrees, by providing them with:

- Background: the necessary technical competencies, including knowledge of scientific principles and skill at rigorous analysis and creative design
- Breadth: a broad education that includes knowledge of current issues and trends in society and technology
- Professionalism: professional attitudes and ethics and skills for clear communication and responsible teamwork
- Learning environment: a learning environment that is rigorous, challenging, open, and supportive

The Computer Science B.S. degree program at UCR is accredited by the Engineering Accreditation Commission of ABET, \[www.abet.org\].

The **Computer Engineering** major stresses the study of core computer science and electrical engineering topics. It prepares students for careers in the design of complex systems involving computer hardware, computer technology to support business processes, and public sectors and for life-long learning, including the option for graduate degrees, by providing them with:

- Background: the necessary technical competencies, including knowledge of scientific principles and skill at rigorous analysis and creative design
- Breadth: a broad education that includes knowledge of current issues and trends in society and technology
- Professionalism: professional attitudes and ethics and skills for clear communication and responsible teamwork
- Learning environment: a learning environment that is rigorous, challenging, open, and supportive

The Computer Engineering B.S. degree program at UCR is accredited by the Engineering Accreditation Commission of ABET, \[www.abet.org\].

The **Business Informatics** major covers the core of computer science and basic business and management topics. It prepares students for careers in design and management of computer and information systems, system and network administration, and e-commerce. It is also useful for careers that apply information technology to support business processes.

The objective of the B.S. degree program in Business Informatics is to prepare graduates for professional practice in both the private and public sectors and for life-long learning, including the option for graduate degrees, by providing them with:

- Background: the necessary technical competencies, including knowledge of scientific principles and skill at rigorous analysis and creative design
- Breadth: a broad education that includes knowledge of current issues and trends in society and technology
- Professionalism: professional attitudes and ethics and skills for clear communication and responsible teamwork
- Learning environment: a learning environment that is rigorous, challenging, open, and supportive

All undergraduates in the College of Engineering must see an advisor at least annually. Visit [student.engr.ucr.edu](http://student.engr.ucr.edu) for details.
University Requirements  
See Undergraduate Studies section.  

College Requirements  
See The Marlan and Rosemary Bourns College of Engineering, Colleges and Programs section.  

The Computer Science major uses the following major requirements toward the satisfaction of some of the college’s Natural Sciences and Mathematics breadth requirements and one of the college’s English Composition breadth requirements.  

1. ENGR 180W  
2. MATH 008B or MATH 009A  
3. PHYS 040A, PHYS 040B, PHYS 040C  

The Business Informatics major uses the following major requirements toward the satisfaction of the college’s Social Sciences breadth requirements and one of the College’s Natural Science and Mathematics breadth requirements.  

1. ECON 002, ECON 003  
2. MATH 008B or MATH 009A  
3. SOC 150  

Major Requirements  

Computer Science Major  

1. Lower-division requirements (61 units)  
a) ENGR 001M  
b) BUS 020  
c) CS 010 or CS 010V, CS 012 or CS 012V, or CS 013, CS 014, CS 061  
d) CS 013/MATH 011  
e) ECON 002, ECON 003  
f) MATH 008B or MATH 009A, MATH 009B, MATH 009C, MATH 010A, and MATH 031  

2. Upper-division requirements (93 units)  
a) ENGR 101M  
b) BUS 103, BUS 104/STAT 104, BUS 106/ ECON 134  
c) CS 100, CS 141, CS 153, CS 165  
d) At least two courses from CS 164, CS 166, CS 172, CS 180  
e) CS 111/MATH 111  
f) ENGR 180W  
g) SOC 150  
h) STAT 155  
i) Sixteen (16) units of upper-division Computer Science technical electives, which must be distinct from the courses used to satisfy the above major requirements. These 16 units may be chosen from those courses listed as upper-division requirements or technical electives for the Computer Science major. At least three courses must be in the Department of Computer Science and Engineering.  
j) Twenty-four (24) units of Business Administration technical electives, including at least 8 units of courses listed in the Information Systems concentration within the Business Administration major. These 24 units must be distinct from the courses used to satisfy the above major requirements and may be chosen from any of the available Business Administration courses, with the following restrictions: no credit will be given for BUS 101, only one of BUS 171 and CS 180 can be taken for credit, only one of BUS 173 and CS 166 can be taken for credit, only one of BUS 175 and CS 164 can be taken for credit, and only one of BUS 125 and CS 177 can be taken for credit.  

Business Informatics Major  

1. Lower-division requirements (56 units)  
a) ENGR 001M  
b) BUS 020  
c) CS 010 or CS 010V, CS 012 or CS 012V, or CS 013, CS 014, CS 061  
d) CS 011/MATH 011  
e) ECON 002, ECON 003  
f) MATH 008B or MATH 009A, MATH 009B, MATH 009C, MATH 010A, and MATH 031  

2. Upper-division requirements (93 units)  
a) ENGR 101M  
b) BUS 103, BUS 104/STAT 104, BUS 106/ ECON 134  
c) CS 100, CS 141, CS 153, CS 165  
d) At least two courses from CS 164, CS 166, CS 172, CS 180  
e) CS 111/MATH 111  
f) ENGR 180W  
g) SOC 150  
h) STAT 155  
i) Sixteen (16) units of upper-division Computer Science technical electives, which must be distinct from the courses used to satisfy the above major requirements. These 16 units may be chosen from those courses listed as upper-division requirements or technical electives for the Computer Science major. At least three courses must be in the Department of Computer Science and Engineering.  
j) Twenty-four (24) units of Business Administration technical electives, including at least 8 units of courses listed in the Information Systems concentration within the Business Administration major. These 24 units must be distinct from the courses used to satisfy the above major requirements and may be chosen from any of the available Business Administration courses, with the following restrictions: no credit will be given for BUS 101, only one of BUS 171 and CS 180 can be taken for credit, only one of BUS 173 and CS 166 can be taken for credit, only one of BUS 175 and CS 164 can be taken for credit, and only one of BUS 125 and CS 177 can be taken for credit.  

Students may petition for exceptions to the above degree requirements. Exceptions to Computer Science course requirements must be approved by the Computer Science and Engineering undergraduate advisor or chair.  

Visit the Student Affairs Office in the College of Engineering or student.engr.ucr.edu for a sample program.  

Minor in Computer Science  

The minor in Computer Science is designed to enhance majors with limited computational theory or practice. As such, students with majors in Computer Engineering, Computer Science, Business Informatics, and Mathematics (Computational Mathematics option) are not eligible.  

Requirements for the minor in Computer Science are:  

1. Lower-division courses: CS 010, CS 012 or CS 013, CS 014, CS 061, CS 011/MATH 011, MATH 008B or MATH 009A, MATH 009B, MATH 009C  
2. Core courses: CS 100, CS 111  
3. Three elective courses, each of four or more units, such that:  
   a) Each is an upper-division requirement or a listed technical elective for the Computer Science major, excluding courses numbered 190-199  
   b) No course may be an upper-division requirement of the student’s major  
   c) At least two courses must be in the Department of Computer Science and Engineering  

4. All courses for the minor must be taken for a letter grade.  

Note: Students with a minor in Computer Science must obtain approval from the undergraduate advisor in Computer Science and Engineering for a specific program of electives consistent with their career goals.  

Graduate Program  

The Department of Computer Science and Engineering offers the M.S. and Ph.D. degrees in Computer Science. General requirements are listed in the Graduate Studies section of this catalog. Specific requirements for each degree are described below.  

Students enrolled prior to Fall 2008 can still follow the old Graduate Program.  

Admission: All applicants must supply GRE General Test scores. The GRE subject test in Computer Science is recommended but not required. Applicants should have at least an undergraduate degree in computer science or a closely related field, but applicants who fail to meet this criterion may sometimes be admitted with deficiencies.  

Prerequisite Material: Competence in the areas defined by the following UCR courses is essential to graduate study in computer science:  

CS 141, CS 150, CS 152, CS 153, CS 161/CS 162, CS 164, CS 165, CS 166, CS 168, CS 169, CS 170, CS 171, CS 172, CS 177, CS 179 (E-Z) (4 units maximum), CS 180, CS 181, CS 182, CS 183, CS 193 (4 units maximum), EE 140, MATH 120, MATH 135A, MATH 135B. The technical electives selected must be distinct from those used to satisfy the requirements specified in 2.a–h) above.  

Visit the Student Affairs Office in the College of Engineering or student.engr.ucr.edu for a sample program.
A student who is deficient in any of these competency areas may be asked to complete the corresponding UCR course with a letter grade of at least B+, or to pass a challenge examination based on that course’s final exam with a grade of at least B+. All such remedial work should be completed within the first year of graduate study, and in all cases the deficiency must be corrected before a student can enroll in any graduate course from the same specialty area.

Core Areas Students have considerable flexibility in selecting specialty area(s) within the program. However, the following core areas introduce fundamental concepts and tools of general interest to all students.

1. Theoretical foundations: CS 203A or CS 220.

Major Specialty Areas The department has active research programs in the following major specialty areas. A list of related graduate courses is provided for each area. Courses that qualify for the M.S. Breadth Requirement are marked with an asterisk (*).

C. Databases, Data Mining, and Machine Learning: CS 205*, CS 235*, CS 229, CS 236*, CS 272
E. Computer Networks: CS 204*, CS 237, CS 239*, CS 240, CS 257, CS 255*
F. Programming Languages, Compilers, and Software Engineering: CS 201*, CS 206*, CS 207*, CS 245*, CS 246*

Master’s Degree

The Department of Computer Science and Engineering offers the M.S. degree in Computer Science, after completion of the following degree requirements.

Satisfactory completion of CS 287 (Colloquium in Computer Science) each quarter of enrollment for full-time in-residence graduate students.

Course Requirements 48 quarter units of graduate or upper-division undergraduate courses are required. Students who have completed similar courses elsewhere may petition for a waiver of a required course or for substitution of an alternative course. For students interested in interdisciplinary research, individual study programs can be approved.

1. Core Requirement (8 units). Choose one course from two of the three Core Areas listed above, with no grade lower than B-.

2. Breadth Requirement (8 units). Two approved breadth courses chosen in such a way that together the core and breadth courses cover four different Major Specialty Areas (A to G).

3. Electives (32 units)
   a. Project Option. A student pursuing the M.S. degree, non-thesis option, may include up to 4 units of Directed Research (CS 297) towards the elective requirement. Of the remaining 28 units, at least 12 units must be approved graduate lecture courses. The remaining 16 units may include additional approved graduate lecture courses, up to 8 units of graduate seminars in CS 260–269, and up to 12 units of approved undergraduate technical electives.
   b. Thesis Option. A student pursuing the M.S. degree, thesis option, may include up to 12 units of graduate research (CS 297 or CS 299) towards the elective requirement. Of the remaining 20 units, at least 4 units must be approved graduate lecture courses. The remaining 16 units may include additional approved graduate lecture courses, up to 8 units of graduate seminars in CS 260–269, and up to 8 units of approved undergraduate technical electives.

Capstone Experience All students must complete a capstone experience that synthesizes and integrates the knowledge and skills obtained throughout the master’s program, according to one of the following options. It is the responsibility of the student to find a faculty member willing to supervise the master’s project or thesis, to form the faculty examining committee, and to schedule the oral examination.

a. Project Option Students must complete a research project under the guidance of a faculty member. This project will require a written report and will be presented to a committee of at least two faculty members in an oral examination.

b. Thesis Option Students must submit a master’s thesis in accordance with the general requirements of the university. The thesis is original research work, and it should demonstrate the student’s ability to study a research area, identify an open problem and make a research contribution. The thesis must be presented to and approved by a committee of at least three faculty members.

Normative Time to Degree 2 years.

Combined B.S. + M.S. Five-Year Program The college offers combined five year B.S. + M.S. programs designed to allow successful UCR Computer Science or Computer Engineering B.S. graduates to complete the Master of Science degree in Computer Science in one year, by allowing up to 12 credits of coursework taken as a UCR undergraduate to be counted towards the 32-unit elective requirements of the M.S. (The courses that can be double counted are those that are eligible to be counted as technical electives in the B.S. requirements.)

A student may apply at the start of their senior year by submitting an application to the Computer Science M.S. program, provided that at the end of junior year, the student was a UCR Computer Science or Computer Engineering B.S. student with cumulative GPA at least 3.4 and had completed the following courses with no grade less than a B- and average grade at least 3.2: CS 100, CS 120A, CS 120B, CS 161. The application to the M.S. program must include at least two recommendation letters from UCR Academic Senate faculty members (at least one, and preferably both, CSE faculty). Submission of GRE scores with the application is recommended but not required. Matriculation into the combined program occurs in the Fall term following senior year, provided: (a) the M.S. application is accepted, (b) throughout senior year, the student is a Computer Science or Computer Engineering B.S. major with cumulative GPA 3.4 or higher, (c) by the end of senior year, the student completes the Computer Science or Computer Engineering B.S. degree requirements.

Incoming students who are applying to the Computer Science or Computer Engineering B.S. programs may simultaneously apply for preliminary admission into the combined program provided their high school GPA is at least 3.6, their SAT-I combined score is at least 1950, they satisfy the Entry Level Writing requirement before matriculation, and they have sufficient math preparation to enroll in calculus upon arrival. Preliminary admission status is maintained as long as the student is in a Computer Science or Computer Engineering B.S. student in good standing with a cumulative GPA of at least 3.4. Preliminarily admitted students still need to apply for full admission in their senior year as described above.

Five-year programs leading to M.S. degrees in other programs (including Computer Engineering) are also available. They are described separately in the catalog sections for those programs.

Doctoral Degree

The Department of Computer Science and Engineering offers the Ph.D. degree in Computer Science, after completion of the following degree requirements. It provides a research-oriented education in preparation for a career in research, industry, or academia and exploring both the fundamental aspects of computer science and engineering as well as their applications.

Satisfactory completion of CS 287 (Colloquium in Computer Science) each quarter of enrollment for full-time in-residence graduate students.

Course Work The course requirements for the Ph.D. degree ensure that Ph.D. students are exposed to fundamental concepts and tools (core requirement), a deep up-to-date view of their research specialty area (depth requirement), and an advanced, up-to-date view of the same topics outside their area (breadth requirement). Students are expected to complete all of these course requirements in the first two years of the program. These
requirements consist of 44 quarter units of approved graduate or upper-division undergraduate courses, satisfying all four of the following course work categories. All of these courses must be taken for a letter grade, and no course can be counted towards more than one category. Students who have completed similar courses elsewhere may petition for a waiver of a required course or for substitution of an alternative course.

Units obtained in CS 270, CS 287, CS 290, CS 297, CS 298, CS 299, CS 301, and CS 302 cannot be counted in any course work category.

1. Core Requirement (12 units). Choose three courses from at least two of the three Core Areas described above, with no grade lower than B- and an overall core course GPA of at least 3.2.

2. Depth Requirement (8 units). Choose two courses listed above under the same Major Area (A to G). This requirement ensures that Ph.D. students, early on in their careers, acquire some depth of knowledge in a particular research area.

3. Breadth Requirement (12 units). Choose three courses from at least two different Major Areas (A to G) outside the student’s depth area. No course that is listed in the student’s depth area can be used to fulfill the breadth requirement, even if it is cross-listed in another area. Students, with the consent of the major professor, may petition for a non-core course to be counted towards the breadth requirement.

4. Electives (12 units). The remaining courses can be selected from additional CS graduate lecture courses, up to 8 units of graduate seminars in CS 260-269, and up to 8 units of approved undergraduate technical electives. Students, with the consent of the major professor, may petition for a non-CS course to be counted as an elective.

Milestones The Department has established three milestones to mark progress towards the Ph.D. degree in Computer Science: advancement to candidacy, presentation of the dissertation proposal, and final oral examination. A Ph.D. student must also satisfy all applicable Graduate Division requirements for each milestone.

Milestone I: Advancement to Candidacy, A student advances to candidacy after he/ she has completed all of the Ph.D. course requirements described above, and passed the combined written and oral qualifying examinations, as described below. These two exams are intended to verify three components of the student's preparation for Ph.D. research: (1) breadth of comprehension sufficient to enable Computer Science research in areas beyond the topic(s) of the research exam and dissertation; (2) ability to perform critical study, analysis and writing in a focused area; and (3) demonstrated research experience or ability to do research.

Written Qualifying Examination The written qualifying examination consists of a written report summarizing the oral presentation to be given at the oral qualifying examination. This report must be written in proper technical English and in the style of a typical Computer Science conference or journal publication, and must be submitted to the Qualifying Committee for approval at least one week prior to the oral qualifying examination.

Oral Qualifying Examination The student is expected to demonstrate research aptitude by undertaking a research study on some topic (typically a problem from student’s chosen research specialty that may be a promising area in which to conduct the dissertation research), under the guidance of his or her faculty major professor. The research must be presented orally to a Qualifying Committee, which is appointed by the Graduate Division based on nominations from the department. The committee will consist of at least four Senate faculty members, with at least three members whose home department is CSE. The committee evaluates the merits of the work and the student’s aptitude for research. The work must represent significant progress towards original and publishable research. The student must complete this requirement in no more than two attempts. The normative time for taking the Oral Qualifying Exam is by the end of the second year.

Dissertation Committee After advancing to candidacy, the student must form a Doctoral Examination Committee chaired by his or her major professor. The committee will consist of at least four Senate faculty members with at least three members belonging to the CSE department (their home department is CSE).

Milestone II: Dissertation Proposal Examination After advancement to candidacy, the student prepares a dissertation proposal that describes the dissertation topic, summarizes the relevant background literature, and presents a comprehensive research plan for the doctoral dissertation. The Dissertation Proposal Examination evaluates appropriateness of the research topic and the feasibility of the research plan. It also establishes a realistic timeline for the completion of the Dissertations. The Dissertation Committee administers this exam. The normative time for the Dissertation Proposal Exam is by the end of the third year. The Dissertation Proposal exam must be taken at least six months prior to the Final Doctoral Examination.

Milestone III: Final Doctoral Examination The student is required to write a dissertation in accordance with the Graduate Division requirements and to defend it in a public oral final doctoral examination to the Dissertation Committee. After a satisfactory performance on the final doctoral examination, the Dissertation Committee recommends granting the Ph.D. degree. The student’s research and the dissertation must both meet the highest standards of originality and scholarship.

Professional Development Requirement All incoming M.S. and Ph.D. students must enroll in the Fall, Winter, and Spring offerings of CS 287, Colloquium in Computer Science.
CS 012V Introduction to Computer Science for Science, Mathematics, and Engineering II (4) Discussion, 2 hours; written work, 6 hours. Prerequisite(s): CS 010 or CS 010V with a grade of "C" or better; familiarity with the C or CS 005 structured and object-oriented programming in C++. Emphasizes good programming principles and development of substantial programs. Topics include recursion, pointers, linked lists, abstract data types, and libraries. Also covers software engineering principles. Uses an online instruction approach. Credit is awarded for only one of CS 012V or CS 012 or CS 013.

CS 013 Introductory Computer Science for Engineering Majors (4) Lecture, 3 hours; laboratory, 3 hours. Prerequisites: CS 012 or CS 012V with a grade of "C" or better; familiarity with C or C++ language. Covers structured and object-oriented programming in C++. Emphasizes good programming principles and development of substantial programs. Topics include recursion, pointers, linked lists, abstract data types, and libraries. Covers software engineering principles. Utilizes examples and assignments specific to engineering disciplines such as numerical data analysis, matrix computations, and dynamic systems. Credit is awarded for only one of CS 012 or CS 012V or CS 013.

CS 014 Introduction to Data Structures and Algorithms (4) Lecture, 3 hours; laboratory, 3 hours. Prerequisite(s): CS 012 or CS 012V with a grade of "C" or better; familiarity with C or C++ language. Covers basic data structures such as arrays, lists, stacks, and queues. Covers dictionaries (including binary search trees and hashing) and priority queues (heaps). Offers an introductory analysis of algorithms, sorting algorithms, and object-oriented programming including abstract data types, inheritance, and polymorphism. Explores solving complex problems through structured software development.

CS 030 Introduction to Computational Science and Engineering (4) Lecture, 2 hours; laboratory, 3 hours. Prerequisite(s): MATH 009C (may be taken concurrently); consent of instructor if credit has been awarded for CS 010 or CS 010V. Examines fundamental programming concepts using the Matlab language including problem decomposition, control structures, elementary data structures, file input/ output, graphics, and code libraries. Focuses on applications problems in engineering and science such as numerical equation solvers; matrix operations; searching and sorting; and data analysis. Emphasizes good programming style and computational efficiency.

CS 049 (E-Z) Language Laboratory (2) For hours and prerequisites, see segment descriptions. Hands-on, direct exposure to a specific programming language and associated development tools in a laboratory setting. Focus is on exercises and practical applications Graded Satisfactory (S) or No Credit (NC). Each segment is repeatable as topics change to a maximum of 12 units.

CS 049E Introductory C and C++ (2) Lecture, 1 hour; laboratory, 3 hours. Prerequisite(s): CS 005 or CS 010 or CS 010V or knowledge of programming or consent of instructor. A practical introduction to software development using C and C++ in a laboratory setting. Focuses on syntax, concepts, selections from the standard library, and development tool-chain use. Graded Satisfactory (S) or No Credit (NC). Segment is repeatable as topics change to a maximum of 12 units.

CS 049M Matlab (2) Lecture, 1 hour; laboratory, 3 hours. Prerequisite(s): CS 005 or CS 010 or CS 010V or knowledge of programming or consent of instructor. A practical exploration of problem solving using Matlab in a laboratory setting. Focuses on syntax, or better CS 012 with a grade of "C" or better. Covers the design and development of digital systems. Topics include Boolean algebra; combinational and sequential logic design; design and use of arithmetic/logic units, carry-lookahead adders, multiplexers, decoders, comparators, multipliers, flip-flops, registers, counters, and standard peripheral devices. Involves use of hardware description languages, synthesis tools, programmable logic, and significant hardware prototyping. Cross-listed with EE 120A.

CS 049Y Python (2) Lecture, 1 hour; laboratory, 3 hours. Prerequisite(s): CS 005 or CS 010 or CS 010V or knowledge of programming or consent of instructor. A practical exploration of problem solving and software development using Python in a laboratory setting. Focuses on syntax, concepts, standard library, and development tool-chain use. Graded Satisfactory (S) or No Credit (NC). Segment is repeatable as topics change to a maximum of 12 units.

CS 061 Machine Organization and Assembly Language Programming (4) Lecture, 2 hours; laboratory, 3 hours. Prerequisite(s): CS 010 or CS 010V with a grade of "C" or better. An introduction to computer organization. Topics include number representation, combinations and permutations, computer instructions, memory organization, addressing modes, interrupt, input/output (I/O), assembly language programming, assemblers, and linkers.

CS 066 Introduction to Three-Dimensional Digital Modeling (4) Lecture, 3 hours; laboratory, 3 hours; individual study, 2 hours. Covers basic skills necessary to create three-dimensional digital images and models. Emphasizes techniques for polygon and curved-surface modeling and photorealistic image creation through material shading, texturing, and lighting. Introduces basic 3D graphic scripting methods to create complex models and images with Autodesk Maya or equivalent. Normally graded Satisfactory (S) or No Credit (NC), but students may petition the instructor for a letter grade. An online hands-on 3D digital modeling training using microcontrollers and simple input/output devices. Covers bit-level manipulation, time-oriented programming using timers, input/output, concurrency, creating a simple task scheduler, task communication, utilization and scheduling, and coding issues. Credit is awarded for only one of CS 121 or CS 121B/ECE 121B.

CS 067 Three-Dimensional Digital Modeling and Animation (4) Lecture, 3 hours; laboratory, 3 hours; individual study, 2 hours. Prerequisite(s): ART 066/ CS 066. Builds advanced skills for three-dimensional modeling. Introduces basic computer animation techniques within framework of existing software. Techniques include rigging skeletons for character models, keyframing, and special effects animation using Autodesk Maya software or equivalent. Teaches proficiency in animation or rendering techniques. Normally graded Satisfactory (S) or No Credit (NC), but students may petition the instructor for a letter grade on the basis of assigned extra work or examination. Course is repeatable to a maximum of 8 units. Cross-listed with ART 067.

Upper-Division Courses

CS 100 Software Construction (4) Lecture, 3 hours; laboratory, 3 hours. Prerequisite(s): CS 014 with a grade of "C-" or better. Emphasizes development of software systems. Topics include design and implementation strategies; selection and mastery of programming languages, environment tools, and implementation strategies; selection and mastery of software development tools and techniques; and significant hardware prototyping. Cross-listed with CS 120A/EE 120A.

CS 108 Introduction to Embedded Systems (5) Lecture, 3 hours; laboratory, 6 hours. Prerequisite(s): CS 100A/EE 120A. Introduction to hardware and software design of digital computing systems embedded in electronic devices (e.g., digital cameras or portable video games). Includes embedded processor programming, custom processor design, and development tool-chain use. Graded Satisfactory (S) or No Credit (NC). Segment is repeatable as topics change to a maximum of 12 units.

CS 121 Programming Embedded Systems (4) Discussion, 2 hours; written work, 6 hours. Prerequisite(s): CS 010 or CS 010V and CS 061, or consent of instructor. An online study of the programming of embedded computing systems involving C language and microcontrollers. Includes an introduction to embedded systems, overview of C programming, bit-level manipulation; and creating time-oriented behavior using synchronous state machines. Covers input/output; concurrency; creating a simple task scheduler; task communication, utilization and scheduling, and coding issues. Credit is awarded for only one of CS 121 or CS 121B/ECE 121B.

CS 121L Laboratory in Programming Embedded Systems (2) Discussion, 1 hour; laboratory, 3 hours; written work, 6 hours. Prerequisite(s): CS 121 (may be taken concurrently), or consent of instructor. An online hands-on embedded systems programming training using microcontrollers and simple input/output devices. Covers bit-level manipulation, time-oriented programming using timers, input/output, concurrency, creating a simple task scheduler, task communication, utilization and scheduling, and coding issues. Credit is awarded for only one of CS 121L or CS 121B/ECE 121B.

CS 122A Intermediate Embedded and Real-Time Systems (3) Lecture, 3 hours; laboratory, 6 hours. Prerequisite(s): CS 012 or CS 012V or CS 013; CS 121/L or CS 120B. Covers software and hardware design of embedded computing systems. Includes hardware and software codesign, advanced programming paradigms (including state machines and concurrent processes), real-time programming and operating systems, basic control systems, and modern chip and design technologies. Laboratories involve use of microcontrollers, embedded microprocessors, programmable logic and advanced simulation, and debug environments.

CS 122B Advanced Embedded and Real-Time Systems (3) Lecture, 3 hours; laboratory, 6 hours. Prerequisite(s): CS 122A. Explores state-of-the-art aspects of building embedded systems. Topics include real-time programming, synthesis of coprocessor cores, application-specific processors, hardware and software cosimulation and co-design, low-power design, reconfigurable computing, core-based design, and platform-based methodology.

CS 130 Computer Graphics (4) Lecture, 3 hours; laboratory, 3 hours. Prerequisite(s): CS 100, MATH 031 (MATH 031 may be taken concurrently); consent of instructor. A study of the fundamentals of computer graphics necessary to design and build graphics applications. Examines raster graphics algorithms including scan-converted graphics primitives, anti-aliasing, and clipping. Also covers geometric transformations, viewing, solid modeling techniques, hidden-surface removal algorithms, color models, illumination, and shading.

CS 133 Computational Geometry (4) Lecture, 3 hours; laboratory, 3 hours. Prerequisite(s): CS 100, CS 111, MATH 031, or equivalents. An introduction to the design of geometry algorithms. Covers the basic computational geometry concepts and techniques used in graphics, robotics, and engineering design. Topics include polygons and polytopes, convex hulls, and voronoi diagrams.
CS 134 Video Game Creation and Design (4) Lecture, 3 hours; laboratory, 3 hours. Prerequisite(s): CS 130. Covers academic, theoretical, and practical aspects of video games by exploring common algorithms, data structures, and design for different genres. Topics include game interface, character movement, intelligent behaviors, and networking or multiplayer games. Requires in-depth, applied programming and a term project, including design, implementation, and analysis of a computer game.

CS 141 Intermediate Data Structures and Algorithms (4) Lecture, 3 hours; laboratory, 3 hours. Prerequisite(s): CS 014 with a grade of "C-" or better; CS 111; MATH 099C or MATH 09HC; proficiency in C++. Explores basic algorithm analysis using asymptotic notations, summation and recurrence relations, and algorithms and data structures for discrete structures including trees, strings, and graphs. Also covers general algorithm design techniques including "divide-and-conquer," the greedy method, and dynamic programming. Integrates knowledge of data structures, algorithms, and programming.

CS 145 Combinatorial Optimization Algorithms (4) Lecture, 3 hours; discussion, 1 hour. Prerequisite(s): CS 141; MATH 031 or MATH 131. The study of efficient algorithms and techniques for combinatorial optimization problems. Topics include shortest paths, minimum spanning trees, network flows, maximum matchings, stable matchings, linear programming, duality, two-person games, algorithmic techniques for integer programming problems, NP-completeness, and approximation algorithms.

CS 150 Automata and Formal Languages (4) Lecture, 3 hours; discussion, 1 hour. Prerequisite(s): CS 014 with a grade of "C-" or better; CS 111; MATH 099C or MATH 09HC. A study of formal languages. Includes regular and context-free languages; computational models for generating these languages such as finite-state automata, pushdown automata, regular expressions, and context-free grammars; mathematical properties of the languages and models; and equivalence between the models. Also introduces Turing machines and decidability.

CS 151 Introduction to Theory of Computation (4) Lecture, 3 hours; discussion, 1 hour. Prerequisite(s): CS 141, CS 150. The study of fundamental questions about the nature of computing. Topics include Turing machines, computability, reductions, complexity theory, complexity classes P and NP, the P=NP problem, NP-completeness, and other time and space complexity classes.

CS 152 Compiler Design (4) Lecture, 3 hours; laboratory, 3 hours. Prerequisite(s): CS 061, CS 100, CS 111, CS 150. Covers the fundamentals of compiler design. Includes lexical analysis, parsing, semantic analysis, compile-time memory organization, run-time memory organization, code generation, and compiler portability issues.

CS 153 Design of Operating Systems (4) Lecture, 3 hours; laboratory, 3 hours. Prerequisite(s): CS 061, CS 100, CS 111, C++ programming proficiency. Covers principles and practices of operating system design. Includes concurrency, memory management, file systems, protection, security, command languages, scheduling, and system performance.

CS 160 Concurrent Programming and Parallel Systems (4) Lecture, 3 hours; laboratory, 3 hours. Prerequisite(s): CS 061, CS 100, CS 111. A study of concurrent and parallel systems. Topics include modular structures and design, interprocess communication, synchronization, failures, persistence, and concurrency control. Also covers atomic transactions, recovery, language support, distributed interprocess communication, and implementation mechanisms. Provides preparation for the study of operating systems, databases, and computer networking.

CS 161 Design and Architecture of Computer Systems (4) Lecture, 3 hours; discussion, 1 hour. Prerequisite(s): CS 120A/EE 120A; concurrent enrollment in CS 161L. A study of the fundamentals of computer systems; includes the performance evaluation of microprocessors; instruction set design and measurements of use; microprocessor implementation techniques including multicore and pipelined implementations; arithmetic; memory hierarchy; and input/output (I/O) systems.

CS 161L Laboratory in Design and Architecture of Computer Systems (2) Lecture, 1 hour; laboratory, 3 hours. Prerequisite(s): CS 120A/EE 120A; concurrent enrollment in CS 161. Covers the design and simulation of computer systems using hardware description language and simulator. Topics include instruction set architecture design; assemblers; datapath and control unit design; arithmetic and logic unit; memory and input/output (I/O) systems; and integration of all parts into a working computer system.

CS 162 Computer Architecture (4) Lecture, 3 hours; laboratory, 3 hours. Prerequisite(s): CS 161 and CS 161L with grades of "C-" or better. The study of advanced processor design. Topics include CPU pipelining, data and control hazards, instruction level parallelism, branch prediction, and dynamic scheduling of instructions. Also covers Very Long Instruction Word (VLIW) processing, multimedia support, design of microprocessors and embedded processors; basic multiprocessor design, shared memory and message passing, and network topologies.

CS 164 Computer Networks (4) Lecture, 3 hours; laboratory, 3 hours. Prerequisite(s): CS 100, CS 111, CS 153. Covers the fundamentals of computer networks. Topics include network architectures, communication protocols, local area networks, UNIX network programming, verification, network security, and performance studies.

CS 165 Computer Security (4) Lecture, 3 hours; laboratory, 3 hours. Prerequisite(s): CS 141, CS 153. Examines the ways in which information systems are vulnerable to security breaches. Topics include attacks; security labels, lattices, and policies; safeguards and countermeasures; intrusion detection; authorization and encryption techniques; networks; digital signatures, certificates, and passwords; privacy issues, firewalls, and spoofing. Trojan horses and computer viruses; CERT Coordination Center; and electronic commerce.

CS 166 Database Management Systems (4) Lecture, 3 hours; laboratory, 3 hours. Prerequisite(s): CS 100, CS 111. Covers the fundamentals of database management systems; relational, network, and hierarchical models; distributed database concepts; query languages; implementation issues; and privacy and security of the database.

CS 168 Introduction to Very Large Scale Integration (VLSI) Design (4) Lecture, 3 hours; laboratory, 3 hours. Prerequisite(s): CS 120A/EE 120A or consent of instructor. Studies integrated circuit fabrication, device characterization, and circuit simulation. Introduces basic design methodology, physical design rules, MOS logic design, and timing and clock schemes. Covers layout generation, subsystem designs, and circuits for alternative logic styles. Also covers design and simulation using hardware description language and CAD tools. Cross-listed with EE 168.

CS 169 Mobile Wireless Networks (4) Lecture, 3 hours; laboratory, 2 hours; extra reading, 1 hour. Prerequisite(s): CS 153 or consent of instructor. Introduces the fundamentals of wireless and mobile networks. Covers wireless channel models, MAC protocols, and network architectures. Also covers cellular, WLAN and ad hoc networks, and routing in multi-hop wireless networks. Includes wireless security and the impact of wireless links on TCP and other transport layer solutions.

CS 170 Introduction to Artificial Intelligence (4) Lecture, 3 hours; discussion, 1 hour. Prerequisite(s): CS 100 with a grade of "C-" or better; CS 111. An introduction to the field of artificial intelligence. Focuses on discrete-value representations and classical planning. Also covers constraint satisfaction and logical inference.

CS 171 Introduction to Machine Learning and Data Mining (4) Lecture, 3 hours; discussion, 1 hour. Prerequisite(s): CS 100; CS 111. Introduces formalisms and methods in data mining and machine learning. Topics include data representation, supervised learning, and classification. Covers regression and clustering, function approximation, and margin-based methods.

CS 172 Introduction to Information Retrieval (4) Lecture, 3 hours; discussion, 1 hour. Prerequisite(s): CS 100; CS 111; EE 114 or STAT 155. Introduces information retrieval (IR) principles and techniques for indexing and searching document collections. Topics include Web search, text processing, ranking algorithms, search in social networks, and search evaluation. Also studies scalability issues in search engines. Satisfies (Ss) or No Credit (NC) grading is not available.

CS 177 Modeling and Simulation (4) Lecture, 3 hours; laboratory, 3 hours. Prerequisite(s): CS 100; CS 111, C++ programming proficiency. Covers validation of random number sequences; concepts in modeling and systems analysis; and conceptual models and their mathematical and computer realizations. Examines simulation modeling techniques, including object-oriented modeling and discrete-event modeling. Emphasizes the use of simulation libraries used with programming languages such as C++.

CS 179 (E-Z) Project in Computer Science (4) For hours and prerequisites, see segment descriptions. Under the direction of a faculty member, student teams propose, design, build, test, and document software and/or hardware devices or systems. Emphasizes professional and ethical responsibilities and the need to stay current on technology and its global impact on economics, society, and the environment.

CS 179E Compilers (4) Discussion, 1 hour; laboratory, 9 hours. Prerequisite(s): CS 141 and CS 152 with grades of "C-" or better; ENGR 180W; 8 additional upper-division units in Computer Science. Covers the planning, design, implementation, testing, and documentation of a compiler-related system. Incorporates techniques from previous related courses. Emphasizes professional and ethical responsibilities; the need to stay current on technology; and its global impact on economics, society, and the environment.

CS 179F Operating Systems (4) Discussion, 1 hour; laboratory, 9 hours. Prerequisite(s): CS 153 with a grade of "C-" or better; ENGR 180W; 8 additional upper-division units in Computer Science. CS 160 is recommended. Covers the planning, design, implementation, testing, and documentation of an operating systems-related system. Incorporates techniques from previous related courses. Emphasizes professional and ethical responsibilities; the need to stay current on technology; and its global impact on economics, society, and the environment.

CS 179G Database Systems (4) Discussion, 1 hour; laboratory, 9 hours. Prerequisite(s): CS 141 and CS 155 with grades of "C-" or better; ENGR 180W; 8 additional upper-division units in Computer Science. Covers the planning, design, implementation, testing, and documentation of a database-related system. Incorporates techniques from previous related courses. Emphasizes professional and ethical responsibilities; the need to stay current on technology; and its global impact on economics, society, and the environment.

CS 179H Computer Graphics (4) Lecture, 3 hours; laboratory, 3 hours. Prerequisite(s): CS 141. Covers computer graphics. Also covers computer science and its global impact on economics, society, and the environment.
CS 170I Network Analysis (4) Discussion, 1 hour; laboratory, 9 hours. Prerequisite(s): CS 141 and CS 164 with grades of "C-" or better; ENGR 180W; 8 additional upper-division units in Computer Science. Covers the planning, design, implementation, testing, and documentation of a network-related system. Incorporates techniques from previous related courses. Emphasizes professional and ethical responsibilities; the need to stay current on technology, and its global impact on economics, society, and the environment.

CS 170J Computer Architecture and Embedded Systems (4) Discussion, 1 hour; laboratory, 9 hours. Prerequisite(s): CS 100, CS 111, CS 122A, and CS 161 with grades of "C-" or better or consent of instructor; ENGR 180W; 8 additional upper-division units in Computer Science. Covers the planning, design, implementation, testing, and documentation of a computer architecture and embedded systems-related system. Incorporates using techniques presented in previous related courses. Emphasizes professional and ethical responsibilities; the need to stay current on technology, and its global impact on economics, society, and the environment.

CS 170K Software Engineering (4) Discussion, 1 hour; laboratory, 9 hours. Prerequisite(s): CS 180; ENGR 180W; 8 additional upper-division units in Computer Science. Covers the planning, design, implementation, testing, and documentation of a software engineering-related system. Incorporates techniques presented in previous related courses. Emphasizes professional and ethical responsibilities; the need to stay current on technology, and its global impact on economics, society, and the environment.

CS 170M Artificial Intelligence (4) Discussion, 1 hour; laboratory, 9 hours. Prerequisite(s): CS 100, CS 111, and CS 170 with grades of "C-" or better; ENGR 180W; 8 additional upper-division units in Computer Science. Covers the planning, design, implementation, testing, and documentation of an artificial intelligence-related system. Incorporates techniques presented in previous related courses. Emphasizes professional and ethical responsibilities; the need to stay current on technology, and its global impact on economics, society, and the environment.

CS 170N Graphics and Electronic Games (4) Discussion, 1 hour; laboratory, 9 hours. Prerequisite(s): CS 130 with a grade of "C-" or better; ENGR 180W; 8 additional upper-division units in Computer Science. Covers the planning, design, implementation, testing, and documentation of a graphics- or electronic game-related system. Incorporates techniques presented in previous related courses. Emphasizes professional and ethical responsibilities; the need to stay current on technology, and its global impact on economics, society, and the environment.

CS 180 Introduction to Software Engineering (4) Lecture, 3 hours; laboratory, 3 hours. Prerequisite(s): CS 100. A study of software engineering techniques for the development, maintenance, and evolution of large software systems. Topics include requirements and specifications, system design and implementation; debugging, testing, and quality assurance; reengineering, project management; software process; tools, and environments.

CS 181 Principles of Programming Languages (4) Lecture, 3 hours; laboratory, 3 hours. Prerequisite(s): CS 081, CS 100, CS 111, CS 150. Covers the principles of programming language design. Includes the study and comparison of several programming languages, their features, and their implementations.

CS 182 Software Testing and Verification (4) Lecture, 3 hours; discussion, 1 hour. Prerequisite(s): CS 100. A study of modern techniques to assess the quality of software artifacts through functional, performance, and reliability testing. Introduces black box and white box testing techniques. Covers the application of modern testing techniques to computer systems, components, subsystems, and entire systems. Also covers verification as a complementary technique to testing.

CS 183 UNIX System Administration (4) Seminar, 3 hours; laboratory, 3 hours. Prerequisite(s): CS 100. Explores the technical aspects of system administration on a Unix system, including advanced Unix. Includes managing system devices, operating system installation, communications, and networking.

CS 190 Special Studies (1-5) Individual study, 3-15 hours. Prerequisite(s): consent of instructor and department chair. Individual study to meet special curricular needs. Course is repeatable to a maximum of 9 units.

CS 191 Seminar in Research Topics in Computer Science and Engineering (1-5) Seminar, 1 hour; laboratory, 9 hours. Prerequisite(s): CS 180 Introduction to Software Engineering (4) or CS 170 with grades of "C-" or better; ENGR 180W; 8 additional upper-division units in Computer Science. Covers the planning, design, implementation, testing, and documentation of a computer science and Engineering-related system. Incorporates using techniques presented in previous related courses. Emphasizes professional and ethical responsibilities; the need to stay current on technology, and its global impact on economics, society, and the environment.

CS 192 Project Design (1-4) Laboratory, 1-6 hours; scheduled research, 1-3 hours; individual study, 1-3 hours. Prerequisite(s): CS 141; consent of instructor. Individual hardware or software design project to include establishment of objectives and criteria, synthesis, analysis, implementation, testing, and documentation. Course is repeatable to a maximum of 8 units.

CS 194 Independent Reading (1-4) Prerequisite(s): consent of instructor. Independent reading in material not covered in class or available in a specific computer science research area. Provides a foundation for pursuit of further studies in special topics in scientific computing. May be taken Satisfactory (S) or No Credit (NC) by students advanced to candidacy for the Ph.D.

CS 194-I Independent Internship in Computer Science and Engineering (1) Seminar, 1 hour; laboratory, 9 hours. Prerequisite(s): CS 141 and CS 164 with grades of "C-" or better; ENGR 180W; 8 additional upper-division units in Computer Science. Covers the planning, design, implementation, testing, and documentation of an artificial intelligence-related system. Incorporates techniques presented in previous related courses. Emphasizes professional and ethical responsibilities; the need to stay current on technology, and its global impact on economics, society, and the environment.

CS 195 Seminar in Research Topics in Computer Science and Engineering (1-5) Seminar, 1 hour; laboratory, 9 hours. Prerequisite(s): consent of instructor and department chair. Individual study to meet special curricular needs. Course is repeatable to a maximum of 8 units.

Graduate Courses

CS 201 Compiler Construction (4) Lecture, 3 hours; outside research, 3 hours. Prerequisite(s): CS 152. Covers theory of parsing and translation. Also addresses compiler construction, including lexical analysis, syntax analysis, code generation, and optimization. May be taken Satisfactory (S) or No Credit (NC) by students advanced to candidacy for the Ph.D.

CS 202 Advanced Operating Systems (4) Lecture, 3 hours; outside research, 3 hours. Prerequisite(s): CS 153. Examines recent developments in operating systems. Also covers multiprocessing, parallel programming, time sharing, scheduling and resource allocation, and selected topics. May be taken Satisfactory (S) or No Credit (NC) by students advanced to candidacy for the Ph.D.

CS 203A Advanced Computer Architecture (4) Lecture, 3 hours; research, 3 hours. Prerequisite(s): CS 161. Covers contemporary computer systems architecture, including stack computers, parallel computers, pipeline processing, database machines, and multiprocessor architecture. Includes evaluation of computer performance. May be taken Satisfactory (S) or No Credit (NC) by students advanced to candidacy for the Ph.D.

CS 203B Advanced Computer Architecture (4) Lecture, 3 hours; research, 3 hours. Prerequisite(s): CS 161. Covers contemporary computer systems architecture, including stack computers, parallel computers, pipeline processing, database machines, and multiprocessor architecture. Includes evaluation of computer performance. May be taken Satisfactory (S) or No Credit (NC) by students advanced to candidacy for the Ph.D.

CS 204 Advanced Computer Networks (4) Lecture, 3 hours; consultation, 1 hour. Prerequisite(s): CS 141 with a grade of "C-" or better, ENGR 180W; 8 additional upper-division units in Computer Science. Covers advanced topics in computer networks, layering, Integrated Services Digital Networks (ISDN), and high-speed networks. Also covers performance models and analysis, distributed systems and datasets, and case studies. May be taken Satisfactory (S) or No Credit (NC) by students advanced to candidacy for the Ph.D.

CS 205 Artificial Intelligence (4) Lecture, 3 hours; discussion, 1 hour. Prerequisite(s): CS 170 or equivalent. Examines knowledge representation and automated reasoning and their use in capturing common sense and expert knowledge. Also addresses predicate and nonmonotonic logics; resolution and term rewriting; reasoning under uncertainty; theorem provers; planning systems; and belief networks. Includes special topics in natural language processing, perception, logic programming, expert systems, and deductive databases. May be taken Satisfactory (S) or No Credit (NC) by students advanced to candidacy for the Ph.D.

CS 206 Testing and Verification Techniques in Software Engineering (4) Lecture, 3 hours; outside research, 1 hour; written work, 1.5 hours. Prerequisite(s): CS 191 with grades of "C-" or better; ENGR 180W; 8 additional upper-division units in Computer Science. Covers the planning, design, implementation, testing, and documentation of a software engineering-related system. Incorporates using techniques presented in previous related courses. Emphasizes professional and ethical responsibilities; the need to stay current on technology, and its global impact on economics, society, and the environment.

CS 207 Advanced Programming Languages (4) Lecture, 3 hours; outside research, 1.5 hours; written work, 1.5 hours. Prerequisite(s): CS 191 with grades of "C-" or better; ENGR 180W; 8 additional upper-division units in Computer Science. Covers advanced programming techniques in object-oriented languages. Also covers concurrency and synchronization issues. May be taken Satisfactory (S) or No Credit (NC) by students advanced to candidacy for the Ph.D.

CS 210 Scientific Computing (4) Lecture, 4 hours. Prerequisite(s): CS 012 or CS 012W; MATH 010A; MATH 031 or equivalent; or consent of instructor. Utilizes scientific computing in a specific computer science research area. Provides a foundation for pursuit of further studies in special topics in scientific computing. May be taken Satisfactory (S) or No Credit (NC) by students advanced to candidacy for the Ph.D.

CS 211 High Performance Computing (4) Lecture, 3 hours; research, 3 hours. Prerequisite(s): CS 191 with grades of "C-" or better; ENGR 180W; 8 additional upper-division units in Computer Science. Covers high performance computing on multicore shared memory computer systems and distributed memory computing clusters. Also covers high performance scientific libraries and computing application development using pthreads, OpenMP, and Message Passing Interface (MPI) parallel file systems. May be taken Satisfactory (S) or No Credit (NC) by students advanced to candidacy for the Ph.D.

CS 213 Multiprocessor Architecture and Programming (4) Lecture, 3 hours; discussion, 1 hour. Prerequisite(s): CS 203A or consent of instructor. Introduces multi-processing, multicomputer architectures, and CC-Numa multiprocessors. Also covers heterogeneous multiprocessors and interconnection networks. May be taken Satisfactory (S) or No Credit (NC) by students advanced to candidacy for the Ph.D.

CS 215 Theory of Computation (4) Lecture, 3 hours; outside research, 3 hours. Prerequisite(s): CS 150. Covers phrase structure grammars and languages; turing machines; relation of languages to automata; solvable and unsolvable problems; and theoretical limitations of computers. Also examines algorithmic complexity theory; polynomial, exponential, and NP classes P, NP, and correctness proofs. May be taken Satisfactory (S) or No Credit (NC) by students advanced to candidacy for the Ph.D.
CS 217 GPU Architecture and Parallel Programming (4) Lecture, 3 hours; discussion, 1 hour. Prerequisite(s): CS 160 with a grade of “C-” or better or consent of instructor. Introduces the popular CUDA based parallel programming framework provided by Nvidia. Covers the basic CUDA memory/threading models. Also covers the common data-parallel programming patterns needed to develop a high-performance computing application. Exercises computational thinking; a broader range of parallel execution models; and parallel programming principles. May be taken Satisfactory (S) or No Credit (NC) with consent of instructor and graduate advisor. May be taken Satisfactory (S) or No Credit (NC) by students advanced to candidacy for the Ph.D. Cross-listed with EE 217.

CS 218 Design and Analysis of Algorithms (4) Lecture, 3 hours; outside research, 3 hours. Prerequisite(s): CS 141. A study of efficient data structures and algorithms for solving problems from a variety of areas such as sorting, searching, selection, linear algebra, graph theory, and computational geometry. Also covers worst-case and average-case analysis using recurrence relations, generating functions, upper and lower bounds, and other methods. May be taken Satisfactory (S) or No Credit (NC) by students advanced to candidacy for the Ph.D.

CS 220 Synthesis of Digital Systems (4) Lecture, 3 hours; outside research, 3 hours. Prerequisite(s): CS 141, 151. Covers the synthesis and simulation of digital systems. Topics include synthesis at the system, behavioral, register-transfer, and logic levels; application-specific processors; simulation; and emerging design methodology. May be taken Satisfactory (S) or No Credit (NC) by students advanced to candidacy for the Ph.D.

CS 223 Reconfigurable Computing (4) Lecture, 3 hours; written work, 3 hours. Prerequisite(s): CS 202 or CS 203A; consent of instructor. Covers reconfigurable computing, a novel computational model that is fast becoming part of the mainstream in high-performance computing. Addresses architectures, software tools and compilers, programming models, and applications. May be taken Satisfactory (S) or No Credit (NC) with consent of instructor and graduate advisor.

CS 229 Machine Learning (4) Lecture, 3 hours; outside research, 3 hours. Prerequisite(s): CS 141, STAT 160A. A study of supervised machine learning that emphasizes discriminative methods. Covers the areas of regression, classification, and clustering. Topics include linear methods, instance-based learning, neural networks, kernel machines, and additive models. May be taken Satisfactory (S) or No Credit (NC) by students advanced to candidacy for the Ph.D.

CS 230 Computer Graphics (4) Lecture, 3 hours; outside research, 3 hours. Prerequisite(s): CS 141 or CS 218; MATH 031 or MATH 131; graduate standing or consent of instructor. Covers advanced topics related to graphics and necessary fundamentals. Includes geometry representations; affine and perspective transformations; rendering with global illumination and other light models; shading and texture mapping; rasterization and anti-aliasing techniques; and hierarchical and keyframe animation. Also includes projects and/or in-depth programming assignments. May be taken Satisfactory (S) or No Credit (NC) by students advanced to candidacy for the Ph.D.

CS 231 Computer Animation (4) Lecture, 3 hours; outside research, 3 hours. Prerequisite(s): CS 130 or CS 230. Covers advanced topics in computer animation, including motion capture; inverse kinematics; and dynamic simulation. Also examines deformable systems and other natural phenomena; facial animation; 3D high-level behavior control; creature evolution; and procedural techniques. May be taken Satisfactory (S) or No Credit (NC) by students advanced to candidacy for the Ph.D.

CS 233 Pen-Based Computing (4) Lecture, 3 hours; discussion, 1 hour. Prerequisite(s): graduate standing or consent of instructor; computer programming experience. Introduces computational techniques for pen-based input devices and issues related to building practical pen-based systems. May be taken Satisfactory (S) or No Credit (NC) with consent of instructor and graduate advisor. Cross-listed with ME 231.

CS 234 Computational Methods for Biomolecular Data (4) Lecture, 3 hours; outside research, 3 hours. Prerequisite(s): CS 111, CS 141 or CS 218; STAT 155 or STAT 160A. A study of computational and statistical methods aimed at automatically analyzing, clustering, and classifying biomolecular data. Includes combinatorial algorithms for pattern discovery; hidden Markov models for sequence analysis; analysis of expression data; and prediction of the three-dimensional structure of RNA and proteins. May be taken Satisfactory (S) or No Credit (NC) by students advanced to candidacy for the Ph.D.

CS 235 Data Mining Techniques (4) Lecture, 3 hours; term paper, 1.5 hours; project, 1.5 hours. Prerequisite(s): CS 141, CS 166; CS 170 is recommended. Provides students with a broad basis in data mining and data analysis. Covers topics such as mining algorithms and tools. Includes clustering, classification, association rules mining, time series clustering, and Web mining. May be taken Satisfactory (S) or No Credit (NC) by students advanced to candidacy for the Ph.D.

CS 236 Database Management Systems (4) Lecture, 3 hours; outside research, 3 hours. Prerequisite(s): CS 141; CS 153 or equivalent; CS 166; or consent of instructor. Covers principles of file systems; architecture of database management systems; data models; and relational databases. Also examines logical and physical design of databases; hardware and software implementation of database systems; and distributed databases (e.g., query processing, concurrencies, recovery). May be taken Satisfactory (S) or No Credit (NC) by students advanced to candidacy for the Ph.D.

CS 237 Advanced Topics in Modeling and Simulation (4) Lecture, 3 hours; research, 3 hours. Prerequisite(s): CS 177. Covers formal computer simulation modeling model validation and verification. Specified Models and differential equation models. Examines current developments in simulation languages. Also addresses integrated model development and its applications to complex, large-scale problems. May be taken Satisfactory (S) or No Credit (NC) by students advanced to candidacy for the Ph.D.

CS 238 Algorithmic Techniques in Computational Biology (4) Lecture, 3 hours; outside research, 3 hours. Prerequisite(s): CS 141 or CS 218. A study of fundamental algorithms for solving combinatorial or computational problems in molecular biology and genomics. Includes sequence alignment and multiple alignment; bio-database search; gene and regulatory signal recognition; DNA sequence assembly; physical mapping; and reconstruction of evolutionary trees. May be taken Satisfactory (S) or No Credit (NC) by students advanced to candidacy for the Ph.D.

CS 239 Performance Evaluation of Computer Networks (4) Lecture, 3 hours; outside research, 3 hours. Prerequisite(s): CS 164. Covers basic and intermediate queuing theory and queuing networks and their application to practical systems. May be taken Satisfactory (S) or No Credit (NC) by students advanced to candidacy for the Ph.D.

CS 240 Network Routing (4) Lecture, 3 hours; outside research, 3 hours. Prerequisite(s): CS 141 or CS 204; CS 164. An in-depth study of routing in computer networks. Examines general principles and specific routing protocols and techniques. Topics include Internet, Asynchronous Transfer Mode (ATM), optical, wireless, and ad hoc networks. May be taken Satisfactory (S) or No Credit (NC) by students advanced to candidacy for the Ph.D.

CS 241 Advanced Topics in Network Measurements and Security (4) Lecture, 3 hours; outside research, 3 hours. Prerequisite(s): CS 164 or equivalent. Introduces measuring and building real network systems. Includes hands-on measurement studies and tools. Covers fundamental mathematical and statistical tools; exposure to implementation studies and techniques; principles of network architectures; and challenges in building testbeds and conducting measurements. Explores measurements and modeling of wireline, ad hoc, sensor, and cellular networks. May be taken Satisfactory (S) or No Credit (NC) by students advanced to candidacy for the Ph.D. Course is repeatable as content changes to a maximum of 8 units.

CS 242 Information Retrieval and Web Search (4) Lecture, 3 hours; term paper, 1.5 hours; project, 1.5 hours per week. Prerequisite(s): CS 141, CS 166. Introduces Information Retrieval (IR) principles and techniques for indexing and searching document collections with special emphasis on Web search. Includes text processing, algorithms, search in social networks, search evaluation, and search engines scalability. May be taken Satisfactory (S) or No Credit (NC) by students advanced to candidacy for the Ph.D.

CS 245 Software Evolution (4) Lecture, 3 hours; research, 3 hours. Prerequisite(s): CS 180 or equivalent; graduate standing. Covers the principles, tools, and techniques for disciplined software evolution. Includes migration strategies, change patterns, software maintenance, legacy system reengineering, reverse engineering, program understanding, middleware, source code analysis, software visualization, and program transformation tools. May be taken Satisfactory (S) or No Credit (NC) by students advanced to candidacy for the Ph.D.

CS 246 Advanced Verification Techniques in Software Engineering (4) Lecture, 3 hours; outside research, 3 hours. Prerequisite(s): CS 111, CS 141, CS 150, or equivalents or consent of instructor. A study of advanced techniques to specify and examine the correctness of complex software systems. Focuses on concurrent and distributed behaviors, formal description languages, temporal logics, model checking and symbolic model checking, partial order reduction, and the use of verification tools. May be taken Satisfactory (S) or No Credit (NC) by students advanced to candidacy for the Ph.D.

CS 253 Distributed Systems (4) Lecture, 3 hours; outside research, 3 hours. Prerequisite(s): CS 153. Integrates the theory and practice of distributed systems with a focus on recent developments in distributed systems. Includes distributed architectures; distributed process management and real-time scheduling; dependable and distributed communication protocols. Also covers distributed process management, replication, large-scale peer-to-peer systems; Internet content delivery; and Web caching. May be taken Satisfactory (S) or No Credit (NC) by students advanced to candidacy for the Ph.D.

CS 255 Computer Security (4) Lecture, 3 hours; outside research, 3 hours. Prerequisite(s): CS 153 or CS 164 or CS 165. Discusses the theoretical and practical issues arising in the context of computer systems security and the principles underlying the design of secure computing environments. Topics include cryptography, secret authentication, public key protocols, network security, intrusion detection, attacks and their countermeasures, and secure systems design. May be taken Satisfactory (S) or No Credit (NC) by students advanced to candidacy for the Ph.D.
CS 257 Wireless Networks and Mobile Computing (4) Lecture, 3 hours; outside research, 3 hours. Prerequisite(s): CS 141; CS 164 or CS 204. Introduces basic and advanced concepts of wireless networks and mobile computing. Covers both wireless cellular and ad hoc networks. Includes protocols for medium access control, resource allocation, and routing, as well as transport layer optimizations for the wireless environment. Also covers standards, Bluetooth, and the IEEE 802.11 for wireless local area networks. May be taken Satisfactory (S) or No Credit (NC) by students advanced to candidacy for the Ph.D.

CS 260 Seminar in Computer Science (4) Seminar, 4 hours. Prerequisite(s): consent of instructor. Covers current research topics in computer science. May be taken Satisfactory (S) or No Credit (NC) by students advanced to candidacy for the Ph.D. Course is repeatable to a maximum of 8 units.

CS 261 Seminar in Artificial Intelligence and the Design of Expert Systems (4) Seminar, 4 hours. Prerequisite(s): graduate standing or consent of instructor. Designed to introduce students to the fields of artificial intelligence and logic programming. Emphasizes expert systems, automated reasoning, and knowledge representation. May be taken Satisfactory (S) or No Credit (NC) by students advanced to candidacy for the Ph.D. Course is repeatable to a maximum of 8 units.

CS 262 Algorithms and Data Structures (4) Seminar, 4 hours. Prerequisite(s): CS 215, CS 218, or consent of instructor. Selected topics in theoretical computer science. May be taken Satisfactory (S) or No Credit (NC) by students advanced to candidacy for the Ph.D. Course is repeatable to a maximum of 8 units.

CS 263 Seminar in Distributed Systems (4) Seminar, 4 hours. Prerequisite(s): graduate standing. CS 153 or previous operating systems course. Introduces the fundamental topics in distributed computer systems. Topics include distributed file systems, replicated data, load management, and distributed shared memory. May be taken Satisfactory (S) or No Credit (NC) by students advanced to candidacy for the Ph.D. Course is repeatable to a maximum of 8 units.

CS 267 Seminar in Databases (4) Seminar, 4 hours. Prerequisite(s): CS 236 or consent of instructor. Focuses on recent research and development issues in the database area. Includes object-oriented databases, heterogeneous databases, parallel databases, benchmarks, transaction processing, query optimization, and performance evaluation. May be taken Satisfactory (S) or No Credit (NC) by students advanced to candidacy for the Ph.D. Course is repeatable to a maximum of 8 units.

CS 270 Seminar in Algorithms (4) Seminar, 4 hours. Prerequisite(s): consent of instructor. Involves presentations and discussions by faculty and students that focus on recent research in computer science. Graded Satisfactory (S) or No Credit (NC). Course is repeatable.

CS 272 Probabilistic Models for Artificial Intelligence (4) Lecture, 3 hours; written work, 3 hours. Prerequisite(s): CS 141, STA 160A. Covers methods for representing and reasoning about probability distributions in complex domains. Focuses on graphical models and their extensions such as Bayesian networks, Markov networks, hidden Markov models, and dynamic Bayesian networks. Topics include algorithms for probabilistic inference, learning models from data, and decision making. May be taken Satisfactory (S) or No Credit (NC) by students advanced to candidacy for the Ph.D.

CS 278 Colloquium in Computer Science (1) Colloquium, 1 hour. Prerequisite(s): graduate standing. Lecture on current research topics in computer science and topics relating to professional development presented by faculty and visitors. Graded Satisfactory (S) or No Credit (NC). Course is repeatable.

CS 290 Directed Studies (1-6) Individual study, 3-18 hours. Prerequisite(s): consent of instructor. Special studies in computer science. Graded Satisfactory (S) or No Credit (NC). Course is repeatable to a maximum of 12 units.

CS 297 Directed Research (1-6) Outside research, 3-18 hours. Prerequisite(s): graduate standing; consent of instructor. Directed research on selected projects in computer science under the sponsorship of assigned faculty members. Graded Satisfactory (S) or No Credit (NC). Course is repeatable to a maximum of 72 units.

CS 298-1 Individual Internship (1-12) Written work, 1-12 hours; internship, 2-24 hours. Prerequisite(s): graduate standing; consent of instructor. Individual apprenticeship in computer science. Includes fieldwork with an approved professional individual or organization and academic work under the direction of a faculty member. A final written report is required. Graded Satisfactory (S) or No Credit (NC). Course is repeatable to a maximum of 16 units.

CS 299 Research for Thesis or Dissertation (1-12) Outside research, 3-36 hours. Prerequisite(s): graduate standing and consent of instructor. Research in computer science under the direction of a faculty member. To be included as part of the thesis or dissertation. Graded Satisfactory (S) or No Credit (NC). Course is repeatable.

Professional Courses

CS 302 Apprentice Teaching (1) Activity, 3 hours. Prerequisite(s): enrollment limited to teaching assistants and associates in Computer Science. Supervised teaching in upper- and lower-division Computer Science courses. Acts in the learning of effective teaching methods such as the handling of Computer Science discussion sections, preparation and grading of examinations, and student relations. Required each quarter of all Computer Science teaching assistants and associates. Graded Satisfactory (S) or No Credit (NC). Course is repeatable to a maximum of 15 units.

Conservation Biology

Subject abbreviation: BLCN
College of Natural and Agricultural Sciences

Program Office, 1223 Pierce Hall
(951) 827-7294; ccb.ucr.edu

The major in Conservation Biology is not currently accepting new students. Students who are interested in this field should see the academic advisors at the CNAS Undergraduate Academic Advising Office, (951) 827-7294.

Upper-Division Courses

BLCN 190 Special Studies (1-4) Individual study, 3-12 hours. Prerequisite(s): consent of instructor and Program Chair. To be taken as a means of meeting special curricular needs. Course content, style, requirements, and grading basis is selected in consultation with the instructor and Program Chair. Course is repeatable to a maximum of 12 units.

BLCN 197 Research for Undergraduates (1-2) Outside research, 3-6 hours. Prerequisite(s): sophomore, junior, or senior standing in Conservation Biology; consent of instructor and Program Chair. An introduction to research providing the opportunity, through reading and preliminary laboratory work, to develop a research project suitable for BLCN 199. Graded Satisfactory (S) or No Credit (NC). Course is repeatable to a maximum of 4 units.

BLCN 198-1 Individual Internship in Conservation Biology (2-4) Internship, 6-12 hours; consultation, 1 hour; outside reading, 2-4 hours. Prerequisite(s): upper-division standing in Conservation Biology. An off-campus practical experience in the public or private sector related to conservation biology that is conducted under the joint supervision of an off-campus sponsor and a faculty mentor from the Conservation Biology Program. A written report on the internship is required. Graded Satisfactory (S) or No Credit (NC). Course is repeatable to a maximum of 12 units.

BLCN 199 Senior Research (1-4) Laboratory, 3-12 hours. Prerequisite(s): junior or senior standing in Conservation Biology; consent of instructor and Program Chair. BLCN 197 is recommended. Research in conservation biology performed under the supervision of a faculty member in the Conservation Biology Program. A written research report is required. Graded Satisfactory (S) or No Credit (NC). Course is repeatable to a maximum of 12 units.

Creative Writing

Subject abbreviation: CRWT
College of Humanities, Arts, and Social Sciences

Andrew Winer, M.F.A., Chair
Department Office, ARTS 129
(951) 827-5424; creativewriting.ucr.edu

Professors
Mike Davis, C.Phil.
Juan Felipe Herrera, M.F.A.
Tom Lutz, Ph.D.
Susan C. Straight, M.F.A.

Professors Emeriti
Steve Minot, Ph.D.
Elidu Martinez, Ph.D.
D. Chuck Whitney, Ph.D.

Associate Professors
Reza Aslan, Ph.D.
Nalo Hopkinson, M.A.
Michael Jayme, M.F.A.
Laila Lalami, Ph.D.

Assistant Professor
Charmaine Craig, M.F.A.
Goldbery Long, M.F.A.

Cooperating Faculty
Stephanie Hammer, Ph.D.

Major

The Creative Writing major offers a series of workshop courses in poetry, fiction, playwriting, screenwriting, and nonfiction as well as reading courses in poetry and fiction presented from a writer’s point of view. They are taught for the most part by poets, fiction writers, and playwrights.

The writing courses are taught as workshops, so that the subject matter (the students’ stories, poems, and plays) is different each time the course is offered. Incoming freshmen and transfer students can apply for a Chancellor’s Performance Award, for up to $4,500. Contact the department office for more information.

University Requirements

See Undergraduate Studies section.

College Requirements

See College of Humanities, Arts, and Social Sciences, Colleges and Programs section.