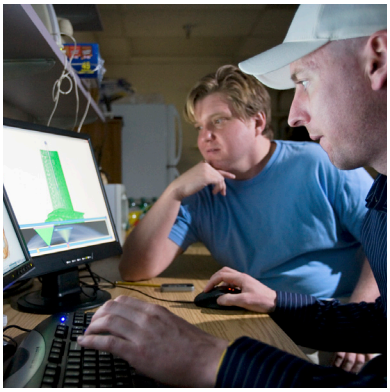
University of New Mexico Department of Computer Science MSC01 1130 Albuquerque, NM 87131

UNM SCHOOL OF ENGINEERING

COMPUTER SCIENCE



2014-2015
GRADUATE INFORMATION PACKET

INTRODUCTION

What is Computer Science?

Computer science is the study of computing systems and computation. This includes theories for understanding computing systems and methods. Computer science is concerned with everything about programs; how to solve problems with them, how to build them, how to classify and analyze them and prove they are correct, how to draw pictures with them, how to get them to learn and much more.

Career Opportunities

Computer science is an exciting and rapidly growing field. Graduates with a Masters degree or a Ph.D. in computer science can expect an entry-level annual salary starting near $100,000. Computer science is a field that will only expand as technology advances, and many wonderful opportunities exist for computer scientists, both during school and after graduation. Graduates might work within the computer industry as systems or application programmer, a systems analyst or designer, or in research. Outside the computer industry, opportunities exist in just about any job or field requiring an expert knowledge of computers and systems of hardware and software.

COMPUTER SCIENCE AT UNM

The CS undergraduate program at UNM is the only Computer Science program in the state accredited by the ABET Computing Sciences Accreditation Board. The Department prides itself on the diversity of its students and faculty, as well as its facilities and opportunities for students at all levels.

Student Opportunities

The Computer Science program at UNM has many opportunities for students to take advantage of. Some students pursue theoretical questions in great depth; others concentrate on diverse practical applications. To help students achieve these and other ambitious goals, the Computer Science Department offers a peer-tutoring program, aggressive nomination of students for awards and fellowships, and departmental scholarships. Through funding from various sources, there are many scholarships available specifically for CS degree candidates, as well as opportunities to connect with research laboratories that open many doors for future careers. Students completing

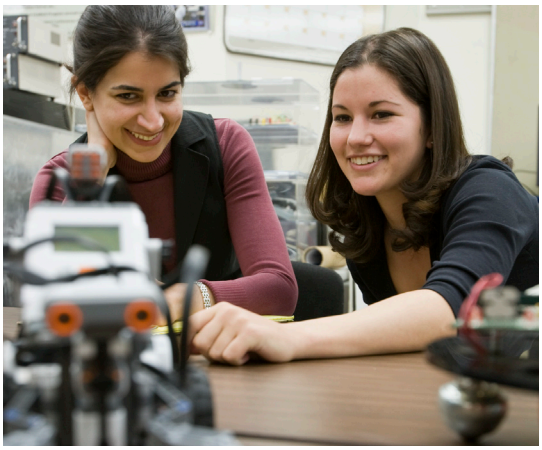
the program have successfully established their own businesses and research groups. The School of Engineering also offers scholarships, and many others are available through the UNM scholarship office. UNM also provides several special programs such as tutoring and counseling, as well as financial assistance for women and minority groups that traditionally have been poorly represented in engineering fields.

Facilities

The UNM Computer Science Department has over 200 general-purpose workstations, running Linux, Windows2000, and Mac OS X. The department network infrastructure consists of a switched 1GB backbone, which links the campus network and supports the principal departmental servers. There is a departmental lab available to students, with 20 workstations, primarily Linux, as well as several labs dedicated to specific research projects. Additional labs are being planned. The Computer Science Department also maintains a wireless network in the Farris Engineering Center for faculty and students. There are two full-time staff and four part-time students providing systems support.

Faculty

In the UNM Computer Science Department, the faculty are committed to working with and assisting students in order to produce graduates who can be effective on the job. Research interests among the faculty include many cutting-edge topics, such as the modeling of avian influenza, computer security, artificial intelligence, high performance computing and arts and technology. In addition, the solid foundation in theory provided in the department ensures both long-term professional competence and the basis for graduate study. Members of the faculty are awarded grants from NSF, DARPA, SFI, DOE and other industry partners, and they work closely with the Santa Fe Institute, Los Alamos National Laboratory and Sandia National Laboratories in order to connect students to opportunities with New Mexico's world-class research.



More Information

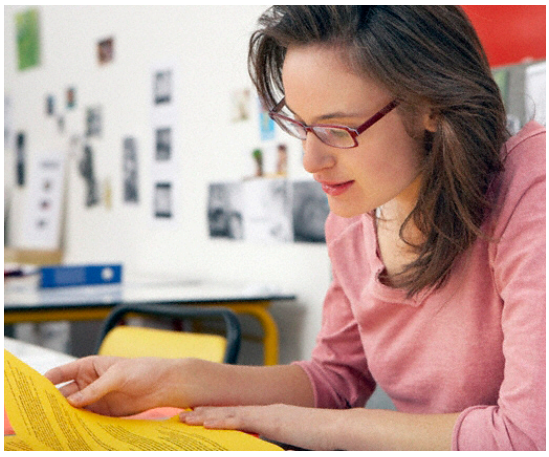
For more information about UNM Computer Science programs, please contact the Department of Computer Science, MSC01 1130, 1 University of New Mexico, Albuquerque, NM 87131-0001, Tel. (505) 277-3112, or send an email to csinfo@cs.unm.edu. You may also visit our website at www.cs.unm.edu.

For general information about the University of New Mexico graduate program, such as admission requirements, application procedures and funding opportunities, please write or call: Office of Graduate Studies – The University of New Mexico, Albuquerque, NM 87131, Tel. (505) 277-2260. You may also visit the Office of Graduate Studies website at http://www.unm.edu/~grad/.

THE GRADUATE PROGRAM

DEGREES OFFERED:
M.S. in Computer Science
Ph.D. in Computer Science

Admission



In addition to the University-wide requirements for admission to graduate study, the prospective M.S. or Ph.D. candidate must submit verbal, quantitative and analytical GRE scores (general test) as well as satisfy the following criteria for admission to graduate study:

1. Knowledge of computer science equivalent to CS 152L, 261, 251L, 341L, 351L, 357L, 361L, 362, 460 and 481.
2. Knowledge of mathematics essential to computer science equivalent to MATH 162, 163, 314 and STAT 345.

Students lacking adequate undergraduate training may be admitted, at the discretion of the admissions committee. The committee will institute a provision requiring deficiencies to be upgraded satisfactorily. Course work required to remove the deficiencies in undergraduate background will not be credited toward the graduate degree. Each student will be assigned a graduate advisor. The student should see his or her graduate advisor before registering for the first time. The student and the advisor will work together to chart out a course of studies which meets the student's career objectives and which constitutes a coherent program satisfying the graduation requirements. The student and the advisor as a part of this coherent program shall count no course toward the required semester hours that has not been agreed upon. It is the responsibility of the student to meet the requirements and to keep the department office informed of compliance with them; in particular, the student should meet with his or her graduate advisor at least once a semester to review progress toward the degree.

Master's Program

In addition to all Office of Graduate Studies requirements for the master's degree, the department also requires the following:

1. Thirty-two semester hours of approved graduate courses.

2. At least 2 semester hours of CS 592 (*Colloquium*), taken at the University of New Mexico.

3. At least 26 of the 32 hours must be in courses offered by the Computer Science

Department at the 500 level or above. Students graduating under Plan I must take a minimum of 6 hours of CS 599 and submit an acceptable thesis. Only 6 hours of CS 599 may be counted toward the 32 hours.

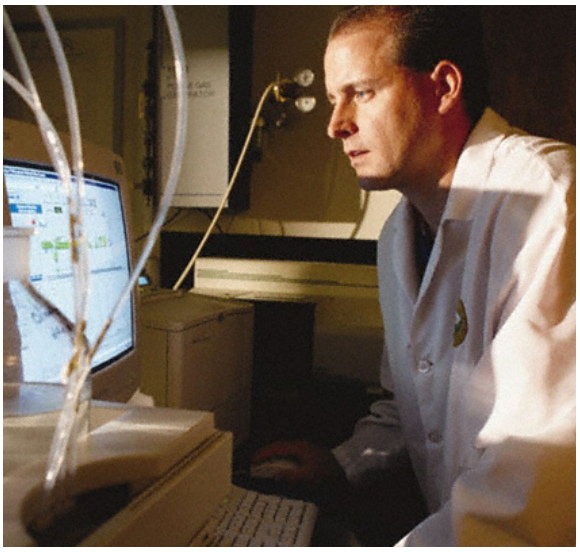
4. Completion of a minimum of two courses from each category below with a B or better grade:
    a. Mathematical methods – CS 500, 530, 550, 558,561.

    b. Empirical methods – CS 512, 522, 523, 527, 529, 547.

    c. Engineering/System Building methods – CS 554, 580, 585, 587.

5. Passing the master's examination. For Plan I students, the master's examination is the defense of thesis. For Plan II students, the master's examination is an oral examination demonstrating mastery of the core areas above.

Doctoral Program



The Ph.D. in Computer Science is offered through a cooperative program involving the Computer Science Departments at the University of New Mexico, New Mexico State University (Las Cruces, NM) and the New Mexico Institute of Mining and Technology (Socorro, NM). Doctoral students at the University of New Mexico may specialize in areas of current interest to the University of New Mexico faculty, or, by special arrangement, they may work in areas of interest to faculty at either of the other two universities.

Requirements

In addition to all Office of Graduate Studies requirements for the Ph.D. degree the department also requires the following:

1. At least 4 semester hours of CS 592 (Colloquium), taken from the University of New Mexico. If the student enters the program with a master's degree, the requirement is reduced to 2 hours of CS 592.
2. At least 24 of the semester hours, exclusive of dissertation, must be completed at one of the three New Mexico universities.
3. At least 30 semester hours, exclusive of dissertation, must be in courses numbered 500 or above. Of these hours, at most 12 may come from individual study courses (at The University of New Mexico, CS 551 and CS 650). If the student enters the program with a master's degree, the requirement is reduced to 18 hours in courses numbered 500 and above – at most 9 of these hours may come from individual study courses.
4. Passing marks on the comprehensive coursework, on the oral candidacy examination and on a final oral examination in the student's area of specialization.

5. Every student who has completed the comprehensive coursework must give one Colloquium per year (scheduled as part of the regular departmental colloquium series) surveying the student's work to date.
6. Teaching requirement for the doctorate: As a requirement for the Ph.D. in Computer Science, all students will complete a one-semester teaching assignment. Typically and preferably, this assignment will involve running a class section, including classroom lecturing. However, there must be some flexibility in tailoring this assignment to each particular student. The student is encouraged to fulfill this requirement early in his or her studies, as the teaching experience is expected to help solidify the student's mastery of core Computer Science material.

Students will progress through three levels of competency. The first is completion of 24 credit hours of coursework from four core areas in Computer Science (Systems, Languages, Theory, and Empirical Methods.

Ph.D. Comprehensive Coursework: complete two courses from each of the following categories with B- or better.

Empirical Methods- CS 512, 523, 527, 529
Languages- CS 550, 558, and choose one of the following 554 or 555
Systems- CS 544, 564, 585, 587
Theory- CS 500, 530, 561

Upon passing the coursework requirement with a 3.5 or higer GPA, the student is allowed to work towards the doctorate.

In the second step, the student's advisor, graduate advisor or department chairperson must then appoint a doctoral committee which will determine the student's remaining program of study and conduct the candidacy examination. The candidacy examination verifies that the student possesses the specialized knowledge required for his/her area of research and ensures that the proposed dissertation topic is adequate in scope, originality and significance. The student is admitted to candidacy for the doctorate upon completion of the comprehensive coursework and candidacy examination, with the approval of the doctoral committee and the Dean of Graduate Studies.

Finally, the committee evaluates the student's doctoral dissertation and conducts the final oral examination on the student's area of specialization. A brochure describing the program and requirements can be obtained from the department.

COMPUTER SCIENCE GRADUATE COURSE OFFERINGS

All prerequisites for graduate level courses require a grad of B or better.

500. Introduction to the Theory of Computation. (3)
Covers basic topics in automata, computability and complexity theory, including: models of computation (finite automata, Turing machines and RAMs); regular sets and expressions; recursive, r.e., and non-r.e. sets and their basic closure properties; complexity classes; determinism vs. non-determinism with and without resource bounds; reductions and completeness; practice with NP- and P-completeness proofs; and the complexity of optimization and approximation problems.
Prerequisite: 401. **NOTE:** CS 401, *Theoretical Foundations of Computer Science*, is primarily for graduate students who are deficient in mathematical proof techniques. This course does not carry graduate credit

506. Computational Geometry. (3)
Development of algorithms and data structures for the manipulation of discrete geometric objects in two- and three-dimensional space. Typical problems include intersection and union of polyhedra, convex hulls, triangulation, point location, neighborhood structures and path computations.
Prerequisite: 561.

510. Randomized Algorithms. (3)
Introduction to probability theory, moments and deviation, tail inequalities, probabilistic method, Markov Chains and Random Walks, dynamic graph algorithms, data stream algorithms, and parallel and distributed algorithms.
Prerequisites: 500 and 530 and 561.

512. Advanced Image Synthesis. (3)
Course covers image synthesis techniques from perspective of high-end scanline rendering including physically-based rendering algorithms. Topics: radiometry, stochastic ray tracing, variance reduction, photon mapping, reflection models, participating media, advanced algorithms for light transport.

513. Real Time Rendering and Graphics Hardware. (3)
Course covers advanced algorithms in real-time rendering and graphics hardware, bringing students up to speed with cutting edge real-time graphics. Topics: advanced GPU algorithms for graphics and non-graphics applications. Term project required.

520. Topics in Interdisciplinary Biological and Biomedical Sciences . (3, no limit) Course covers varying interdisciplinary topics taught by collaborative scientists from UNM, SFI, and LANL.

522./422. Digital Image Processing. (3)
Introduction to fundamentals of digital image processing. Specific topics include grey level histograms, geometric/grey level transformations, linear systems theory, Fourier transforms, frequency domain filtering, wavelet transforms, image compression, edge detection, color vision, and binary image morphology.
Prerequisites: 351L and MATH 314.

523. Complex Adaptive Systems. (3)
A graduate introduction to computational tools to measure, simulate and analyze complexity in biological and social systems. Topics include cellular automata, dynamical systems, genetic algorithms and other biologically inspired computational methods. Programming maturity is required.

524. Collaborative Interdiciplinary Teaching. (3)
Course designed to develop the methods content and assessment of effective interdisciplinary biological courses; Students will develop and teach an undergraduate interdisciplinary topics course. Topics vary.
Restriction: permission of instructor.

527./427. Principles of Artificially Intelligent Machines. (3)
Survey of artificial intelligence exclusive of pattern recognition. Heuristic search techniques, game playing, mechanical theorem proving, additional topics selected by the instructor.
Prerequisite: 351L.

528. Advanced Topics in Artificial Intelligence. (3)
Continues the topics presented in 427/527, including writing an expert system shell in LISP; designing and building an object-oriented interpreter; creating a hybrid environment by attaching rules to objects. Representation issues to include: semantic nets, frames, objects, conceptual graphs and others. Assignments include writing a recursive descent semantic net parser.
Prerequisite: 427 or 527.

529./429. Introduction to Machine Learning. (3)
Introduction to principles and practice of systems that improve performance through experience. Topics include statistical learning framework, supervised and unsupervised learning, Bayesian analysis, time series analysis, reinforcement learning, performance evaluation and empirical methodology; design tradeoffs.
Prerequisite: 362 or 530 or 561.

530. Geometric and Probabilistic Methods in Computer Science. (3)
Introduction to applied mathematics for computer scientists. Specific topics include discrete and continuous random variables (including transformation and sampling), information theory, Huffman coding, Markov processes, linear systems theory, Fourier transforms, principal component analysis, and wavelet transforms.
Prerequisite: STAT 345.

531. Pattern Recognition. (3)
(Also offered as ECE 517.) Decision functions and dichotomization; prototype classification and clustering; statistical classification and Bayes theory; trainable deterministic and statistical classifiers. Feature transformations and selection. Introduction to sequential, hierarchical and syntactic methods.
Prerequisites: STAT 345 or ECE 340.

532. Computer Vision. (3)
(Also offered as ECE 516.) Theory and practice of feature extraction, including edge, texture and shape measures. Picture segmentation; relaxation. Data structures for picture description. Matching and searching as models of association and knowledge learning. Formal models of picture languages.
Prerequisites: STAT 345 or ECE 340, CS 361L or ECE 331.

544./444. Introduction to Cyber Security. (3)
This class will focus on proactive security, i.e. designing networks, algorithms and data structures which are provably robust to attack. Grades will be based on class participation, presentations, and class projects.
Prerequisite: 561 and (362 or 500) with a grade of B or better.

547. Neural Networks. (3)
(Also offered as ECE 547.) A study of neuron models, basic neural nets and parallel distributed processing.
Prerequisite: MATH 314 or 321.

550. Programming Languages and Systems. (3)
Current trends in design and philosophy of languages and systems. Data abstraction, data flow languages, alternative control structures, environments, correctness, software tools.
Prerequisite: 451.

551. Individual Study–Graduate. (1-3 to a maximum of 6)
Guided study, under the supervision of a faculty member, of selected topics not covered in regular courses. Restriction: permission of instructor.

554./454. Compiler Construction. (3)
Syntax analysis and semantic processing for a block-structured language. Lexical analysis, symbol tables, run-time management. Students will write a compiler.
Prerequisites: 341L and 351L.

555. Advanced Topics in Compiler Construction. (3)
Aspects needed to write production quality compilers. Optimization, error recovery, parse table compression, semantic processing of complex data structures, type checking, runtime support, code generation, compiler-writing systems.
Prerequisite: 454 or 554.

557. Selected Topics in Numerical Analysis. (3, no limit.)
(Also offered as MATH 557.) Possible topics include approximation theory, two point boundary value problems, quadrature, integral equations and roots of nonlinear equations. May be repeated for credit.

558. Software Foundation.
Introduction to modern programming techniques and programming language features and the theory used to describe and define programming languages, using types as the organizational principle.

561. Algorithms/Data Structure. (3)
Study of data structures and algorithms and mathematics needed to analyze their time and space complexity. Topics include: amortized analysis and self-adjusting data structures for trees and priority queues, graphing algorithms, greedy and divide-and-conquer paradigms.
Prerequisite: 362.

564./464. Introduction to Database Management. (3)
Introduction to database management systems. Emphasis is on the relational data model. Topics covered include query languages, relational design theory, file structures and query optimization. Students will implement a database application using a nonprocedural query language interfaced with a host programming language.
Prerequisite: 561.

565. Topics in Database Management. (3)
A continuation of 464/564 with emphasis on query optimization, leading-edge data models, transaction management and distributed databases. Additional topics determined by student interests.
Prerequisite: 564.

573./473. Physics and Computation. (3) Moore
(Also offered as NSMS 573) A survey of complex systems at the interface between physics and computer science, including phase transition, power laws, social networks, NP-completeness, and Monte Carlo methods.

575. Introductory Numerical Analysis: Numerical Linear Algebra. (3)
(Also offered as MATH 504.) Direct and iterative methods of the solution of linear systems of equations and least squares problems. Error analysis and numerical stability. The eigenvalue problem. Descent methods for function minimization, time permitting.
Prerequisites: MATH 464 or MATH 514. {Spring}

576. Introductory Numerical Analysis: Approximation and Differential Equations. (3)
(Also offered as MATH 505.) Solution of nonlinear problems and minimization. Numerical approximation of functions. Interpolation by polynomials, splines and trigonometric functions. Numerical integration and solution of ordinary differential equations. An introduction to finite difference and finite element methods, time permitting.
Prerequisites: MATH 316 or MATH 401. {Fall}

580. The Specification of Software Systems. (3)
A comparative study of the techniques used to specify software systems. The course will emphasize formal techniques and will cover the specification of sequential and concurrent systems. Although no programming will be required, students will be required to write specifications for several small software systems.
Prerequisite: 460.

581. Fundamentals of Software Testing. (3)
Introduces the components of software development life cycle and role of software test process, test planning and strategy, static testing, tracking bugs, dynamic testing, use of automated testing as well as other testing concepts.

583. Object-Oriented Testing. (3)
Introduction to software test process. Topics include: testing perspectives, object-oriented concepts, UML diagrams, development/testing processes, test design, test case development, verifying tests, test case automation, test patterns, and understanding basic concepts of class hierarchies.

585. Computer Networks. (3)
A theoretical and practical study of computer networks, including network structures and architectures; protocols and protocol hierarchies; error handling; routing; reliability; point-to-point networks; broadcast networks; local area networks; efficiency and throughput; communications technologies; case studies.

587. Advanced Operating Systems. (3)
Theory of design of operating systems. Modeling, simulation, synchronization, concurrency, process hierarchies, networks and distributed systems.

590. Topics in Computer Science for Non-Majors–Graduate. (1-3, no limit)
This course is intended to provide students in other disciplines with an opportunity to study aspects of modern computer science, tailored to their own field of study. May be repeated for credit.
Restriction: permission of instructor.

591. Special Topics–Graduate. (1-6, no limit)
Graduate seminars in special topics in computer science. May be repeated for credit.
Restriction: permission of instructor.

592. Colloquium. (1 to a maximum of 4)
Required of all graduate students. May be repeated, with at most 2 credits towards the M.S. requirements and at most 2 further credits towards the Ph.D. requirements. Students will write a short essay on the topic of one or more of the colloquia offered that semester.
Offered on a CR/NC basis only.

599. Master's Thesis. (1-6, no limit)
Offered on a CR/NC basis only. May be repeated for credit.

650. Reading and Research. (3 to a maximum of 6) Restriction: permission of instructor.
May be repeated for credit.

691. Seminar in Computer Science. (1-6 to a maximum of 12)
Restriction: permission of instructor. May be repeated for credit.

699. Dissertation. (3-12, no limit)
Offered on a CR/NC basis only. May be repeated for credit.