

COMPUTER SCIENCE DEPARTMENT

The Computer Science Department at William and Mary offers a stimulating, collegial environment in which to pursue a Master's or Ph.D. degree. With coursework in several broad areas of computer science, William and Mary provides a wide spectrum of advanced study and research opportunities. Defining qualities include the opportunity for easy interaction with faculty, and equal dedication to research and teaching – a tradition rooted in the university's history as the nation's second-oldest institution of higher learning. Graduate students benefit from the proximity of NASA Langley Research Center, and the Thomas Jefferson National Accelerator Facility (TJNAF), institutions which offer opportunities for collaborative research.



Established in 1986, the graduate program features an excellent placement record for its graduates. Our master's students have found employment primarily with major computer system manufacturers, software development companies, and within the aerospace and defense industry. Our Ph.D. students have gone on to tenure-track academic positions, or have

accepted industrial research and development positions. The department's current graduate enrollment is about ninety students. Historically, one-third of the graduate students have been women.

The Master's (M.S.) program is appropriate for students who would like to improve their professional competence or prepare for future doctoral study. The program can normally be completed in two years or less without prior graduate-level coursework. The department also offers, in conjunction with the Mathematics Department, a specialized M.S. degree in Computational Operations Research (COR). COR students conduct research in modeling real world systems using a variety of mathematical and computational techniques. Sample applications include airline crew scheduling, actuator placement in flexible space structures, allocation of spare parts, job shop scheduling and performance analysis.

Students from either M.S. program can continue for the Ph.D. in Computer Science. The Ph.D. program prepares students for careers in research or academia. The program can generally be completed in five years or less of graduate study and requires a dissertation based on original research that makes a significant contribution to scholarly knowledge in the student's research area.

Areas of Research and Study

The Computer Science Department provides a strong research program with faculty actively engaged in research in the following areas: algorithms, computer systems and networking, high performance computing, modeling and simulation, programming languages and compilers, software verification, software engineering, graphics, and scientific computing.

Faculty

CHAIR **Virginia Torczon** Professor (Ph.D., Rice).

GRADUATE DIRECTOR **Evgenia Smirni** Professor (Ph.D., Vanderbilt).

PROFESSORS **Andreas Stathopoulos** (Ph.D., Vanderbilt), and **Weizhen Mao** (Ph.D., Princeton).

ASSOCIATE PROFESSORS **Phil Kearns** (Ph.D., Virginia), **Peter Kemper** (Ph.D., Dortmund), **Qun Li** (Ph.D., Dartmouth), **Haining Wang** (Wilson P. and Martha Claiborne Stephens Term Distinguished Associate Professor) (Ph.D., Michigan), and **Xipeng Shen** (Ph.D., Rochester).

ASSISTANT PROFESSORS **Isil Dillig** (Ph.D., Stanford), **Thomas Dillig** (Ph.D., Stanford), **Pieter Peers** (Ph.D., K.U. Leuven), **Denys Poshyvanyk** (Ph.D., Wayne State), and **Gang Zhou** (Ph.D., Virginia).

RESEARCH ASSISTANT PROFESSOR **Saskia Mordijck** (Ph.D., University of California, San Diego).

VISITING ASSOCIATE PROFESSOR **Rance Necaie** (Ph.D., William and Mary).

INSTRUCTOR **Deborah S. Noonan** (M.S., William and Mary).

PROFESSOREMÉRITI **William L. Bynum** (Ph.D., North Carolina), **Stefan Feyock** (Ph.D., Wisconsin), **Robert Noonan** (Ph.D., Purdue), **Richard H. Prosl** (Ph.D., Rensselaer), and **Paul K. Stockmeyer** (Ph.D., Michigan).

ADJUNCT PROFESSORS **Rex K. Kincaid** (Ph.D., Purdue), and **Lawrence M. Leemis** (Ph.D., Purdue).

ADJUNCT ASSOCIATE PROFESSOR **R. Michael Lewis** (Ph.D., Rice).

General Description

The Department offers a Master of Science (M.S.) in computer science and a Doctor of Philosophy (Ph.D.) in computer science. In conjunction with faculty from the Mathematics Department, the department also offers a M.S. with a specialization in computational operations research. As part of the Computational Science Cluster, the department offers a M.S. and a Ph.D. with a specialization in computational science. Well qualified students who earn a M.S. with a specialization in computational operations research are encouraged to apply for admission to the Ph.D. program. Faculty are actively engaged in research in the following areas:

- Operating systems: networks, architecture, parallel and distributed computation, security, compiler construction, software engineering;
- Modeling and Simulation: discrete-event simulation, stochastic systems;
- Computational Science: serial and parallel numerical methods, numerical optimization;
- Foundations: analysis of algorithms and graph theory;
- Operations Research: nonlinear programming, discrete optimization, metaheuristics, reliability, computational statistics, stochastic optimization.

Some faculty and graduate students participate in joint research activities with two nearby national research facilities: the NASA Langley Research Center and the Thomas Jefferson National Accelerator Facility (Jefferson Lab).

The department maintains several heterogeneous networks to support teaching and research. These networks include personal computers, Unix workstations, and multiprocessors. Additional information about the department can be found on the web at <http://www.wm.edu/computerscience>.

Admission

Applicants must submit test results for the aptitude portion of the Graduate Record Examination and are encouraged (but not required) to submit results from a suitable subject area. Students from non-English speaking countries must submit TOEFL results.

Admission requirements for the M.S. and Ph.D. in computer science

Students seeking the M.S. or Ph.D. degrees in computer science are expected to have a background that includes the following:

Mathematics: two calculus courses and one linear algebra course.

Computer Science: two introductory programming courses (CS1 and CS2 in the standard computer science curriculum) and one course in each of discrete mathematics, data structures, algorithms, and computer organization.

Applicants lacking this background may be admitted provisionally into the M.S. program. In that case, the department will establish a suitable set of qualifying courses at the time of admission. To achieve regular status, provisionally accepted students must earn at least a B in each qualifying course. There is no provisional admission into the Ph.D. program.

Admission requirements for the M.S. with a specialization in computational operations research

Students seeking the M.S. degree with a specialization in computational operations research are expected to have a background in mathematics, science or engineering and the ability to program in a high-level language. Students with insufficient background in computer science may be required to enroll in CSCI 241 and CSCI 303. Applicants lacking an appropriate background may be admitted provisionally.

Admission requirements for the M.S. with a specialization in computational science

Students seeking the M.S. degree with a specialization in computational science are expected to have a background in mathematics, science or engineering that includes the following:

Mathematics: two calculus courses and one linear algebra course

Computer Science: two introductory programming courses (CS1 and CS2 in the standard computer science curriculum).

Students with insufficient background in data structures, algorithms, computer organization, and systems programming may be required to enroll in CSCI 241, CSCI 303, CSCI 304, and CSCI 315.

Degree Requirements for the Master of Science

(See general College requirements in the section entitled 'Graduate Regulations' in this catalog.)

Students who have taken twelve or more credits in courses leading to a graduate degree must maintain a minimum grade point average of 3.0. Students with less than a 3.0 average may appeal in writing to the department's Committee on Admission, Retention, and Financial Aid to remain in the graduate program. The committee normally rejects appeals from students with less than a 2.75 grade point average.

Students may choose to write a thesis or not. Students who do not choose the thesis option must complete 32 graduate credits, including CSCI 710, Research Project. Students who choose the thesis option must complete 24 graduate credits in addition to CSCI 700, M.S. Thesis, and defend their thesis at an oral examination, open to the faculty and to whomever else the department may invite. CSCI 710, Research Project cannot be applied to the 24 credits. In either case, students can apply at most 12 credits in courses numbered below 600 and must satisfactorily complete CSCI 653 and two of the additional 600 level

courses, excluding 670, 690, 695, and all courses with numbers 6x8.

A student cannot use courses taken in another department, nor CSCI courses designated as Computational Operations Research courses (last digit in the course number is an 8), to satisfy degree requirements without prior written approval of the student's advisor and the Graduate Admissions Committee.

Students must submit a 2-year plan of study at the beginning of their first semester in the M.S. program, which must be endorsed by their faculty advisors and then approved by the Graduate Admission Committee.

Degree Requirements for the Master of Science with a Specialization in Computational Operations Research

Students must complete 32 graduate credits, including one of either CSCI 698, Simulation and Modeling Project in Computational Operations Research, CSCI 708, Research Project in Computational Operations Research, or CSCI 710, Research Project. There is no thesis option for this specialization. In addition, students must satisfactorily complete at least seven courses from the following list of courses in the computational operations research area: CSCI 608, 618, 628, 638, 648, 658, 668, 678, 688, 698, and 708. Math 501, Math 524, and/or Math 552 may be taken and will count towards the required 32 graduate credits.

Degree Requirements for the Master of Science with a Specialization in Computational Science

Students have to abide by the minimum grade point requirements and may choose to write a thesis or not, as specified in the requirements for the Master of Science. In either case, students may apply at most 12 credits in courses numbered below 600 and must satisfactorily complete CSCI 653, CSCI 649, and one additional 600 level course, excluding 670, 690, 695, and all courses with numbers 6x8. In addition students must also satisfactorily complete at least one graduate course from outside the Computer Science Department. Each student will have a two-person computational science advisory committee within the department to advise the student about what is needed to meet the certification requirements of the Computational Science Cluster.

Degree Requirements for the Doctor of Philosophy

(See general College requirements in the section entitled 'Graduate Regulations' in this catalog.)

Students seeking the Ph.D. in computer science must complete an eight course requirement for credit, with at least a 3.7 grade point average in seven of the eight courses, and with no individual grade lower than B-. All eight courses must be taken in the Computer Science Department at William and Mary.

Six courses must be chosen from 600 level courses, excluding 670, 690, 695 and all courses with numbers 6x8. The remaining two courses must be chosen from 700 level courses, excluding 700 M.S. Thesis, 708 Research Project in Computational Science, 710 Research Project, 770 Colloquium, 790 Readings, and 795 Research.

A student may not use courses taken in another department nor CSCI courses designated as Computational Operations Research courses (last digit in the course number is an 8), to satisfy degree requirements without prior written approval of the student's advisor and the Graduate Admissions Committee.

For the Degree of Doctor of Philosophy with a specialization in computational science, the student must satisfy all of the department's requirements for the degree of Doctor of Philosophy. In support of the specialization in computational science, the student must take the CSCI 649 course, and at least one graduate course from outside the department.

If two or more graduate courses outside the department are taken, only one of them can be used to satisfy the department's eight-course requirement. In all cases, the grade point average will be based on the seven courses taken inside the department.

Each computational science student will have a three-person computational science advisory committee within the department to advise the student about what is needed to meet the certification requirements of the Computational Science Cluster. The committee will approve the graduate course(s) from outside the department, and ensure that the dissertation topic incorporates computation in a creative way, either by developing an enabling computational technology, or by using such technologies to obtain a significant scientific result.

Doctoral students must complete a year of continuous residence as a full-time student at William & Mary. Students who obtain a M.S. or M.A. degree must complete their residency requirement after satisfying the requirements for a M.S. or M.A. degree (at William and Mary or elsewhere). Students who do not obtain a M.S. or M.A. degree must complete their residency requirement after satisfying the department's eight course requirement. There is no foreign language requirement.

In addition to required course work, doctoral students will identify a principal research advisor, form a doctoral advisory committee, and petition the department for acceptance into candidacy for the Ph.D. degree. After acceptance into candidacy, students must pass the thesis proposal exam. This examination is oral, is conducted by the candidate's committee, and is open to the faculty and to whomever else the department may invite. Approximately six months before the anticipated dissertation defense, the candidate is required to meet with the committee. At this meeting, the candidate is expected to describe in detail the status of the research upon which the dissertation is based and plan for conducting the work that remains to be done. The purpose of this meeting is to provide the committee with an opportunity to evaluate the candidate's work and plans, and to provide feedback and advice in advance of the defense. The committee may require, at its discretion, additional meetings before a defense date can be scheduled. Candidates must submit and satisfactorily defend a dissertation to a committee of at least five members, with at least one member from outside the department. The dissertation is based on original research and should contribute to the discipline's body of knowledge. The defense is oral and is open to the faculty and to whomever else the department may invite. Each year, the faculty will review how well doctoral students have progressed toward completion of their Ph.D. degree. The department provides written guidelines to help students judge their own progress. In addition, the department provides more specific regulations than those conveyed in this catalog. Students are solely responsible for familiarizing themselves with all guidelines and regulations of the department. In Computer Science and Computer Science with a specialization in Computational Science, students receiving regular admission to the M.S./Ph.D. program have (7) seven years from the time they enter the graduate program to complete all degree requirements. There is no provisional admission to the Ph.D. programs in Computer Science. In Computer Science with a specialization in Computational Operations Research, students receiving admission to the M.S. program in Computational Operations Research must apply to the Ph.D. program in Computer Science after completing the M.S. degree requirements. Such students would have (7) seven years from the time of this second admission to complete the Ph.D. requirements.

Description of Courses

Wherever a William and Mary course is specified as a prerequisite, it is understood that appropriate experience or an equivalent course, taken at another institution, may be substituted for the specified prerequisite. Each such substitution must be approved by the instructor of the course for which a substitution is appropriate. Generally, graduate students should also consult with their advisors to verify that they meet all course prerequisites.

Note that 500-level courses are cross-listed as 400-level courses or 300-level courses in the undergraduate catalog and will thus be open to undergraduates. In these cross-listed courses, there will be higher expectations and additional requirements for graduate students. Students should consult with the instructor of such a course for further information.

Unless otherwise noted, all courses are graded using standard grading [A, B, C, D, F] scheme (See VI. Grading and Academic Progress in the section entitled 'Graduate Regulations' in this catalog) and may not be repeated for credit (See Repeated Courses requirements in the section entitled 'Graduate Regulations' in this catalog).

503. Algorithms.

Spring (3) Prerequisites: Data Structures, Discrete Structures.

A systematic study of algorithms and their complexity, including searching, sorting, selecting, and algorithms for graphs. A survey of algorithm design methods, including greedy algorithms, divide-and-conquer, dynamic programming, and backtracking. An introduction to NP-complete problems. No credits earned in this course may be applied to the number of credits required for a graduate degree. [Cross-listed with CSCI 303]

504. Computer Organization.

Fall (3) Prerequisites: CSCI 241 and CSCI 243.

Organization of computer hardware and software; virtual machines, computer systems organization, machine language, assembler language, and microprogramming. No credit earned in this course may be applied to the number of credits required for a graduate degree. [Cross-listed with CSCI 304]

512. Web Programming.

Spring (3) Prerequisite: CSCI 321. May be taken for Audit.

Overview of the Internet. Markup languages: HTML, CSS, XML. Server-side programming languages: Perl/Python, PHP, Java. Other topics include: N-tier programming, security, database access, XML processing.

515. Systems Programming.

Spring (3) Prerequisite: Computer Organization.

The design and implementation of programs which provide robust and efficient services to users of a computer. Macro processors; scripting languages; graphical interfaces; network programming. Unix and X are emphasized. [Cross-listed with CSCI 315]

520. Elementary Topics.

Fall or Spring (1, 2, or 3 credits, depending on material) Prerequisites: Will be published in the registration schedule. This course may be repeated for credit.

A treatment of elementary topics of interest not routinely covered by existing courses. Material may be chosen from various areas of computer science.

521. Implementation of Database Systems.

Fall (3) Prerequisite: An introductory course in database.

Issues involved in designing efficient database systems, and the strategies, data structures, and algorithms used in the implementation of such systems. Some advanced topics covered: data warehousing, online analytical processing, data mining, spatial data management.

523. Finite Automata and Theory of Computation.

Fall (3) Prerequisites: Linear Algebra, Algorithms.

Theory of sequential machines and finite automata. Turing machines, recursive functions, computability of functions.

524. Computer Architecture.

Fall (3) Prerequisite: Computer Organization.

An introduction to the principles of computer design. Topics include data representation, including adders, signed integer arithmetic, floating point representation and character representation. A study of microprocessor, minicomputer and mainframe architecture including clocks, memory management, bus communication and input/output.

526. Simulation.

Fall (3) Prerequisites: Calculus, Algorithms.

Introduction to simulation. Discrete and continuous stochastic models, random number generation, elementary statistics, simulation of queuing and inventory systems, Monte Carlo simulation, point and interval parameter estimation. Selected applications.

527. Computer Graphics.

Fall (3) Prerequisites: Linear Algebra, Algorithms, Computer Organization.

Introduction to computer graphics and its applications. Topics include coordinate systems, the relationship between continuous objects and discrete displays, fill and flood algorithms, two-dimensional geometric transformations, clipping, zooming, panning, and windowing. Topics from three-dimensional graphics include representations for objects, geometric and projection transformations, geometric modeling, and hidden line/surface removal algorithms.

530. Computer Languages.

Fall and Spring (1 or 2 credits, depending on material) Prerequisites: Will be published in the preregistration schedule.

Topics include syntax, semantics, and pragmatics of one computer language as well as aspects of that language's intended areas of application which influenced its design. The language studied will vary; students may repeat the course for different languages.

534. Network Systems and Design.

Spring (3) Prerequisites: Systems Programming, or permission of instructor.

The Internet; principles and design of network applications, including web servers and multimedia; transport, network and data link layers; network security; network performance evaluation and capacity planning.

535. Software Engineering.

Spring (3) Prerequisites: Programming Languages.

The software life cycle. Software design methodologies. Testing and maintenance. Programming teams.

542. Compiler Construction.

Fall (3) Prerequisites: Algorithms, Computer Organization, Programming Languages.

Principles and tools for the construction of translators for programming languages. Topics include lexical analysis, block structure, grammars, parsing, error recovery, program representation, run-time organization and code generation.

544. Principles of Operating Systems.

Fall (3) Prerequisites: Algorithms, Computer Organization, Systems Programming.

The conceptual view of an operating system as a collection of concurrent processes; semaphores, monitors, and rendezvous. Real and virtual memory organization, resource allocation, file organization and management, processor allocation and management, and external device management.

554. Computer and Network Security.

Spring (3) Prerequisite: CSCI 315.

An introduction to the principles and practices of cryptography, network security, and secure software. Cryptography topics include: basic methods, key distribution and protocols for authenticated and confidential communications. The practice of network security includes: Kerberos, PGP, public key infrastructures, SSL/TLS, IP security, intrusion detection, password management, firewalls, viruses and worms, and Denial of Service (DoS) attacks.

597. Problems in Computer Science.

Fall, Spring and Summer (1) Graded Pass/Fail.

Supervised projects selected to suit the needs of the graduate student, including those wishing to perform an internship as part of the Curricular Practical Training Program. Projects to be chosen in consultation with the student's advisor. Acceptable research outlines and project reports are required. Students may count credits received in only one offering of this course toward the number of credits required for their degree.

608. Decision Theory.

(3) Prerequisite: Equivalent of MATH 351.

Development and use of systematic procedures for assisting decision makers in evaluating alternative choices. Emphasis is on problem formulation, uncertainty and risk assessment, Bayes, minimax and other decision rules and applications. Problems will be solved using appropriate software tools.

616. Stochastic Models in Computer Science.

Fall or Spring (3) Prerequisites: Discrete Mathematics, Calculus.

An introduction to stochastic models, problem solving, and expected value analysis as applied to algorithms and systems in computer science. Topics include probability, discrete and continuous random variables, discrete-time Markov chains, and continuous time birth-death processes.

618. Models and Applications in Operations Research.

(3) Prerequisite: Equivalent of MATH 323.

A study of realistic and diverse Operations Research problems with emphasis upon model formulation, interpretation of results and implementation of solutions. Topics include applications of linear programming, goal programming, decomposition of large-scale problems, and job scheduling algorithms. Problems will be solved using appropriate software tools.

626. Data Analysis and Simulation.

Fall or Spring (3) Prerequisites: Some knowledge of probability and statistics.

Basic statistical analysis techniques for experimental data generation and collection, aiming at design, analytic modeling and implementation of systems. Covers basics from the areas of statistics, simulation, event queueing, and their application to Internet systems, data centers and cloud computing, storage systems, distributed systems, and hardware/software design.

628. Linear Programming.

Fall (3) Prerequisite: Equivalent of MATH 211. Co-requisite: equivalent of CSCI 241.

Theory and applications of linear programming. Topics include the simplex method, duality theory, sensitivity analysis and interior point methods. Problems will be solved using appropriate software tools.

634. Advanced Computer Networking.

Fall or Spring (3) Prerequisites: Computer Networks, or permission of the instructor.

Covers various aspects of computer networking: Internet design principles, wireless, mobile, and sensor networks, MAC protocols, routing, congestion/flow control, network topology and traffic analysis, network security, web service, and overlay networks.

635. Advanced Software Engineering.

Fall or Spring (3) Prerequisites: an undergraduate course in software development.

Covers a range of topics that challenge today's software development teams: the design of large systems, the necessity to adjust and maintain existing software systems over a longer than expected life cycle, the urge for correctness, robustness and performance of software.

638. Nonlinear Programming.

(3) Prerequisites: CSCI 628 and the equivalent of MATH 212.

Topics include unconstrained optimization, nonlinear least-squares, feasible-point methods, and penalty and barrier methods, with an emphasis on effective computational techniques.

643. Automated Logical Reasoning.

Fall or Spring (3) Prerequisite: Knowledge of algorithms and finite automata.

Automated logical reasoning has enabled substantial progress in many fields of computer science, including software and hardware verification, theorem proving, program analysis, and artificial intelligence. In this course, we will study widely-used logical theories and decision procedures for answering whether formulas in these theories are satisfiable. In particular, we will consider automated reasoning techniques for propositional logic, first-order logic, linear arithmetic over reals and integers, theory of uninterpreted functions, and combinations of these theories. This course will examine automated logical reasoning both from a theoretical and practical perspective, giving interested students a hands-on experience building useful tools, such as SAT solvers.

648. Network Optimization.

(3) Prerequisite: CSCI 628.

Network flow theory and algorithms, including transportation, maximum flow shortest path and minimum spanning tree problems. Applications to a variety of areas are also stressed. Problems will be solved using appropriate software tools.

649. Computational Methods.

Fall or Spring (3) Prerequisites: undergraduate Calculus and Linear Algebra.

Covers a wide spectrum of numerical algorithms and techniques for solving real world computational problems. Topics include non-linear and linear equations, interpolation, integration, differentiation, and the central effects of floating point arithmetic. Both theory and programming aspects are covered.

652. Advanced Compiler Construction.

Fall or Spring (3) Prerequisites: Finite Automata, Compiler Construction.

A course on compiler technology with focus on program code optimizations, generation, and parallelization. Topics include data flow analysis, code transformations, dependence theory, parallelization and vectorization, register and cache management, and interprocedural analysis.

653. Analysis of Algorithms.

Fall or Spring (3) Prerequisite: Algorithms.

Algorithm design techniques including divide-and-conquer, dynamic programming and greedy method. Analysis methods including worst case and average case. Additional topics chosen from among amortized analysis, lower bound theory and NP-completeness.

654. Advanced Computer Architecture.

Fall or Spring (3) Prerequisite: Computer Architecture.

A study of high performance computer architecture with emphasis on experiments and simulation. Topics include pipelining, memory hierarchies, I/O, multiprocessors, and new designs for performance improvements.

658. Discrete Optimization.

Spring (3) Prerequisites: CSCI 628 and the equivalent of CSCI 303.

Topics include relaxation techniques, constructive heuristics, improving search techniques (simplex method simulated annealing, tabu search), branch and bound schemes, and valid inequalities for branch and cut methods. Problems will be solved using appropriate software tools.

663. Theory of Computation.

Fall or Spring (3) Prerequisites: Finite Automata and a strong mathematical background.

An in depth study of Turing machines and the equivalent computational models such as recursive function theory and lambda calculus. Church's thesis and incompleteness results. Computational complexity including NP-completeness.

664. Advanced Operating Systems.

Fall or Spring (3) Prerequisite: Operating Systems.

Advanced topics in the design and implementation of modern operating systems, especially those which support a distributed computer environment. Topics include: synchronization, mutual exclusion, language support, process and thread management, scheduling, remote procedure call, fault tolerance, network and parallel file systems, security, modeling and performance.

668. Reliability.

(3) *Prerequisites: equivalent of MATH 401 and CSCI 141.*

Introduction to probabilistic models and statistical method used in analysis of reliability problems. Topics include models for the lifetime of a system of components and statistical analysis of survival times data. Problems will be solved using appropriate software tools.

670. Colloquium.

Fall and Spring (1,1) Graded Pass/Fail. This course may be repeated for credit.

Each full-time graduate student is required to enroll in this course. No credits earned in this course may be applied to the number of credits required for a degree. This course includes training in the responsible and ethical conduct of research, as well as discussions of fabrication, falsification, and plagiarism.

678. Statistical Analysis of Simulation Models.

(3) *Prerequisites: equivalent of MATH 351, MATH 401 and CSCI 141.*

This course introduces statistical techniques used in the analysis of simulation models. The first half of the course develops techniques for determining appropriate inputs to a simulation model, and the last half develops analysis techniques that are applied to the output of a simulation model.

680. Topics.

Fall and Spring (1, 2, or 3 credits, depending on the material covered) Prerequisite: Will be published in the preregistration schedule. This course may be repeated for credit.

A treatment of Master's level topics of interest not routinely covered by existing courses. Material may be chosen from various areas of computer science.

688. Topics in Computational Operations Research.

(3) *May be repeated for different topics. This course may be repeated for credit.*

A treatment of Master's level topics of interest not routinely covered by existing courses. Material may be chosen from various areas of computational operations research.

690. Readings in Computer Science.

Fall or Spring (1, 2, or 3 credits, depending on the material covered) Graded Pass/Fail. Prerequisite: Permission of the instructor and the Chair. This course may be repeated for credit.

A description of the intended contents of the readings course must be approved by the Chair before the student may register for the course. Students electing to satisfy M.S. requirements by taking 24 credits and writing a thesis may not count credits received in this course toward the required 24. Students electing to satisfy M.S. requirements by taking 32 credits may count credits received in only one offering of this course toward the required 32.

695. Research.

Fall and Spring (Hours and credits to be arranged) Graded Pass/Fail. This course may be repeated for credit.

Required of all full-time students who have not attained Research Graduate status. No credits earned in this course may be applied to the number of credits required for a degree.

698. Simulation and Modeling Project in Computational Operations Research.

Fall (3) Prerequisites: equivalent of MATH 401 and CSCI 241.

Simulation model building in a high-level simulation language (SIMAN) with C++/C interface. Topics include network, discrete-event, and continuous modeling approaches. Interfaces between the three modeling approaches are presented. Familiarity with univariate and multivariate probability distributions is required for input modeling and simulation output analysis. Course culminates in a semester project in SIMAN.

700. M.S. Thesis.

Fall and Spring (Hours and credits to be arranged) This course may be repeated for credit.

708. Research Project in Computational Operations Research.

Fall and Spring (2,2) Graded Pass/Fail. Prerequisite: Permission of Graduate Director.

Students will select a faculty advisor and committee in their area of specialization within computational operations research, prepare a research proposal abstract for approval by the department's director of graduate studies, undertake a research project, and write a paper describing their research. This course is normally taken after a student has completed 18 credit hours toward the M.S. degree with a specialization in computational operation research. Not open to students who receive credit for either CSCI 700 or CSCI 710.

710. Research Project.

Fall and Spring (2,2) Graded Pass/Fail. Prerequisites: Permission of Graduate Director.

Students will select a faculty advisor and committee in their area of research interest, prepare a research proposal abstract for approval by the department's director of graduate studies, undertake a research project, and write a paper describing their research. This course is normally taken after a student has completed 18 credit hours toward the M.S. degree. Not open to students who receive credit for either CSCI 700 or CSCI 708.

712. Advanced Compiler Construction II.

Fall or Spring (3) Prerequisites: CSCI 542, CSCI 652.

A project-oriented course involving compilers or compiler generators. Possible topics include syntactic error recovery, semantic analysis, code optimization and code generation.

723. Advanced Analysis of Algorithms.

Fall or Spring (3) Prerequisites: CSCI 653 or consent of the instructor.

Advanced aspects of the design and analysis of computer algorithms. The study of probabilistic algorithms and parallel algorithms for solving problems from graph theory, geometry, and number theory. Lower bound theory. Intractability theory and its application to modern cryptography.

726. Discrete Event Simulation.

Fall or Spring (3) Prerequisites: CSCI 616, CSCI 626.

Methods of discrete-event simulation. Markov chains. Simulation of open and closed networks of queues. Simulation of non-stationary Poisson processes. Transient and steady-state analysis. Event list algorithms and data structures. Theoretical and empirical tests of randomness. Selected applications.

734. Distributed Computing Systems.

Fall or Spring (3) Prerequisites: CSCI 544 or equivalent.

Time and order in distributed systems. Synchronous and asynchronous systems. Models of faulty behavior in distributed systems. Paradigms of distributed computing: network mutual exclusion, deterministic agreement (Byzantine and fail-stop), elections, global state acquisition, atomic transactions. Issues in programming distributed systems. Reliable distributed systems. Distributed databases. Selected case studies.

746. Discrete-State Stochastic Models.

Fall or Spring (3) Prerequisites: CSCI 616, CSCI 626.

Logic, performance, and reliability analysis of discrete-state systems. Exploration of the state space. Queuing networks, fault trees, reliability block diagrams, task graphs, Petri nets and domain-oriented languages. Underlying stochastic processes, solutions and approximations.

749. Numerical Algorithms.

Fall or Spring (3) Prerequisites: CSCI 649, or permission of the instructor.

In-depth study of modern numerical algorithms central to solving many scientific and engineering problems, and of the techniques used to develop and analyze those algorithms, with an emphasis on algorithmic issues.

754. Performance Evaluation of Computer Systems.

Fall or Spring (3) Prerequisites: CSCI 526, CSCI 626, or permission of the instructor.

Analytical modeling techniques and their application in computer system performance modeling and prediction. Modeling of resource allocation policies in parallel systems, web server analysis, measurements and workload characterization of parallel computations and multimedia applications, hardware/software design, and bottleneck analysis.

770. Colloquium.

Fall and Spring (1,1) Graded Pass/Fail. This course may be repeated for credit.

Each full-time graduate student is required to enroll in this course. No credits earned in this course may be applied to the number of credits required for a degree. This course includes training in the responsible and ethical conduct of research, as well as discussions of fabrication, falsification, and plagiarism.

774. Parallel Computing.

Fall or Spring (3) Prerequisites: CSCI 652, CSCI 653, or permission of the instructor.

This course introduces parallel computation as a means of achieving high performance in modern parallel architectures. A unified approach is followed, where the design of parallel algorithms, their implementation and performance evaluation is studied in relation to the underlying system.

780. Advanced Topics.

Fall or Spring (1, 2, or 3 credits, depending on material) Prerequisites: Will be published in the preregistration schedule. This course may be repeated for credit.

A treatment of doctoral-level topics of interest not routinely covered by existing courses. Material may be chosen from various areas of computer science.

790. Readings in Computer Science.

Fall or Spring (1, 2, or 3 credits, depending on the material covered) Graded Pass/Fail. Prerequisites: Permission of the instructor and the Graduate Director. This course may be repeated for credit.

A description of the intended contents of the readings course must be approved by the Graduate Director before the student may register for the course. Students electing to satisfy M.S. requirements by taking 24 credits and writing a thesis may not count credits received in this course toward the required 24. Students electing to satisfy M.S. requirements by taking 32 credits may count credits received in only one offering of this course toward the required 32.

795. Research.

Fall and Spring (Hours and credits to be arranged) Graded Pass/Fail. This course may be repeated for credit.

Required of all full-time students who have not attained Research Graduate status. No credits earned in this course may be applied to the number of credits required for a degree.

800. Doctoral Dissertation.

Fall and Spring (Hours and credits to be arranged) This course may be repeated for credit.