Department of Computer Science

Stuart Building
10 W. 31st St.
Chicago, IL 60616
312.567.5150
312.567.5067 fax
www.iit.edu/csl/cs
info@cs.iit.edu

Chair:
Xian-He Sun

Associate Chair:
Bogdan Korel

The study of computer science is the inquiry into the nature of computation and its use in solving problems in an information-based society. Computer science is an evolving discipline, but it has a well-defined core of knowledge and a set of characteristic methodologies. The methods and skills required of the computer scientist include formalization and abstraction, algorithm design, programming, organization of unstructured knowledge, modeling, language development, and software system architecture and design. The graduate program in computer science at IIT stresses high achievement in both fundamental knowledge and practical problem solving. It offers the student a solid background in the core areas and exposure to cutting-edge computer technologies.

Degrees Offered

Master of Computer Science (MAS)
Master of Computer Science with specialization in:
  - Business
  - Computational Intelligence
  - Cyber-Physical Systems
  - Data Analytics
  - Database Systems
  - Distributed and Cloud Computing
  - Education
  - Finance
  - Information Security and Assurance
  - Networking and Communications
  - Software Engineering

Master of Science in Computer Science
Doctor of Philosophy in Computer Science

With the Department of Applied Mathematics:
Master of Data Science

With the Department of Electrical and Computer Engineering:
Master of Telecommunications and Software Engineering

With the Department of Chemical and Biological Engineering:
Master of Science in Computer Science/Master of Chemical Engineering

Certificate Programs

Computational Intelligence
Cyber-Physical Systems
Data Analytics
Database Systems
Distributed and Cloud Computing
Information Security and Assurance
Networking and Communications
Software Engineering

Research Facilities

The department has research computing facilities that include several state of the at computer clusters and workstations. The equipment includes a large-scale Sun “ComputerFarm” consisting of 172 processors and 562 cores connected via a Linux-based IBM cluster, an Opteron cluster from Microsoft, and a Cray XD1 connected Management. An advanced technology Access Grid node has been installed by the department, which allows researchers, teachers, and students at different locations worldwide to interact via a network multimedia environment. Research in Data-Intensive Distributed Systems is facilitated by a 12-node cluster with 118 cores, 382 GB RAM, and 32TB of harddisk space. Research labs in the department have advanced computer workstations with multicore servers.

Research Areas

Algorithms, data structures, artificial intelligence, computer architecture, computer graphics, computer networking and telecommunications, computer vision, database systems, distributed and parallel processing, I/O systems, image processing, information retrieval, natural language processing, software engineering, and system software, machine learning, cloud computing.
Faculty

Agam, Gady, Associate Professor. B.S., M.S., Ph.D., Ben-Gurion University (Israel). Computer vision, computer graphics, image processing, pattern recognition, machine learning, geometric modeling, medical imaging, document imaging.

Argamon, Shlomo, Professor. B.S. Carnegie-Mellon University; M.S, Ph.D. Yale University. Machine Learning, computational linguistics, stylistics, information retrieval.


Bauer, Matthew J., Senior Lecturer and Director of Undergraduate Academic Advising. B.S., M.S., Illinois Institute of Technology.

Beckman, Jr., A. Mattox, Senior Lecturer. B.S., Ph.D., University of Illinois, Urbana-Champaign.

Bilgic, Mustafa, Assistant Professor. B.S., University of Texas; M.S., Ph.D., University of Maryland. Data mining, machine learning, probabilistic graphical models, statistical relational learning, active learning, social network analysis, information visualization.

Burnstein, Ilene, Professor Emerita. B.S., Brooklyn College; M.S., University of Maryland; Ph.D., Illinois Institute of Technology. Software engineering, knowledge-based testing and debugging tools, test process assessment and improvement models, capability maturity models.

Calinescu, Gruia, Associate Professor. M.S., University of Bucharest (Romania); Ph.D., Georgia Institute of Technology. Algorithms, approximation algorithms, optical and wireless, ad-hoc networks.

Chlebus, Edward, Industry Associate Professor. M.S., Ph.D., Cracow University (Poland). Network modeling, performance evaluation and tele-traffic analysis.

Culotta, Aron, Assistant Professor. B.S., Tulane University; M.S., Ph.D., University of Massachusetts. Social media analysis, information extraction, data mining, machine learning, natural language processing.

Evens, Martha, Professor Emerita. A.B., Bryn Mawr College; A.M., Radcliffe College; Ph.D., Northwestern University. Natural language processing, expert systems and intelligent tutoring/information systems.

Glavic, Boris, Assistant Professor. M.Sc., RWTH Aachen University (Germany); Ph.D., University of Zurich (Switzerland). Databases, data provenance, information integration.

Greene, Peter, Professor Emeritus. A.B., Amherst College; Ph.D., University of Chicago. Neural networks, feeling-based reasoning, artificial intelligence and robotics.

Hanrath, Jon, Senior Instructor. Ph.D., Michigan State University.

Hood, Cynthia, Associate Professor of Computer Science and Engineering. B.S., Rensselaer Polytechnic Institute; M.S., Stevens Institute of Technology; Ph.D., Rensselaer Polytechnic Institute. Network management, statistical signal processing, learning processing.

Jin, Dong, Assistant Professor. B.Eng., Nanyang Technological University (Singapore); M.S., University of Illinois, Urbana-Champaign. Cyber Security, Networks, Modeling and Simulation of Large-Scale Systems, Trustworthy Power Critical Infrastructures.

Kapoor, Sanjiv, Professor. Ph.D., University of Illinois, Urbana-Champaign. Computational geometry, graph algorithms, combinatorial optimization, graphics, data structures.

Korel, Bogdan, Associate Professor of Computer Science and Engineering and Associate Chair, Department of Computer Science. M.S., Technical University of Kiev (Ukraine); Ph.D., Oakland University. Software engineering, automated software testing and analysis.

Koutsogiannakis, George, Instructor. B.S., M.S., M.B.A., DePaul University; M.S. Illinois Institute of Technology.

Lan, Zhiling, Associate Professor. B.S. Beijing Normal University (China); M.S. Chinese Academy of Sciences; Ph.D., Northwestern University. Parallel and distributed computing, performance analysis and modeling.


Li, Xiang-Yang, Professor. B.E., Tsinghua University; M.S., Ph.D., University of Illinois, Urbana-Champaign. Algorithm design and analysis, system design for wireless ad hoc and sensor networks, network information theory, security protocols, and computational geometry.

Raicu, Ioan, Assistant Professor. B.S., M.S., Wayne State University; M.S., Ph.D., University of Chicago. Distributed Systems, high-throughput and high-performance computing, efficient task dispatch and execution systems, resource provisioning, data management, scheduling, performance evaluations in distributed systems.

Reingold, Edward M., Professor. B.S., Illinois Institute of Technology; M.S., Ph.D., Cornell University. Analysis of algorithms, data structures, combinatorial algorithms, mathematics, history, and computer implementation of calendars.
Ren, Shangping, Associate Professor. Ph.D., University of Illinois, Urbana-Champaign. Domain specific computing (including distributed computing, real-time computing, and mobile computing), and domain specific programming abstractions (such as language abstractions for real-time systems, for context-aware adaptive mobile systems, and for coordination among distributed asynchronous entities).

Roberson, Dennis A., Research Professor and Vice Provost for Research. B.S., Washington State University; M.S., Stanford University.

Saelee, Michael K., Senior Instructor. B.S., M.S., Illinois Institute of Technology.

Sasaki, James T., Senior Lecturer and Director of the Professional Master's Program in Computer Science. B.S., Illinois Institute of Technology; M.S., Ph.D., Cornell University.

Sun, Xian-He, Professor and Chair. Ph.D., Michigan State University. Distributed and parallel processing, software systems, I/O systems, performance evaluation, scientific computing.

Wan, Peng-Jun, Professor. B.S., Tsinghua University (China); Ph.D., University of Minnesota. Interconnection design, routing and resource management in optical networks, low-earth orbit satellite networks, wireless local area networks.

Winans, Vida J., Senior Instructor and Graduate Coordinator. B.A., Cornell University; M.S., Illinois Institute of Technology.

Zhang-Sun, Hong, Research Professor. B.S., Beijing Normal University; M.S., Ph.D., Michigan State University.
Admission Requirements

Cumulative undergraduate GPA minimum: 3.0/4.0
Cumulative Master of Science GPA minimum (for Ph.D. applicants): 3.5/4.0
GRE minimum combined (quantitative/verbal/analytical) score for tests taken on or after Oct. 1, 2011:
M.S.: 298 (quantitative + verbal), 3.0 analytical writing
M.A.S.: 292 (quantitative + verbal), 2.5 analytical writing
Ph.D.: 304 (quantitative + verbal), 3.5 analytical writing, with a minimum in the 70th percentile of the quantitative section
GRE minimum score for tests taken on or after Aug. 1, 2002:
M.S.: 1000 (quantitative + verbal) 3.0 analytical writing
MAS: 900 (quantitative + verbal) 2.5 analytical writing
Ph.D.: 1100 (quantitative + verbal) 3.5 analytical writing, with a minimum in the 70th percentile of the quantitative section
TOEFL score (international students from non-English speaking countries): 523/70*
PTE score (international students from non-English speaking countries): 47
IELTS score (international students from non-English speaking countries): 5.5

Meeting the minimum GPA and test score requirements does not guarantee admission. Test scores and GPA are just two of several important factors considered. Applicants to masters degree programs in computer science should hold a bachelors degree in computer science with a minimum overall GPA of 3.0/4.0 or its equivalent. For international students from non-English speaking countries, a minimum TOEFL score of 523/70 is required. All applicants must submit scores from the GRE general test. (The GRE requirement is waived for applicants to the Master of Computer Science program who hold bachelors degrees from accredited U.S. institutions with a minimum cumulative GPA of 3.0/4.0.) Applicants with bachelors degrees in other disciplines can be admitted to Master of Science or Master of Computer Science programs. However, students whose training does not include the equivalent of CS 201 (Accelerated Introduction to Computer Science), CS 330 (Discrete Structures), CS 331 (Data Structures and Algorithms), CS 350 (Computer Organization and Assembly Language Programming) and CS 351 (Systems Programming) will be required to complete all of the courses in which a deficiency exists. Some students may be able to complete their deficiencies with the following six-credit hour sequence with grades of "B" or better:

CS 401 Introduction to Advanced Studies I
CS 402 Introduction to Advanced Studies II

In addition, students who have not had at least one course in calculus will be required to take a calculus course.

Applicants to the Ph.D. program should hold an M.S. degree in computer science with a minimum GPA of 3.0/4.0 for their bachelors degree and 3.5/4.0 for their M.S. degree, or a minimum GPA of 3.5/4.0 for their bachelors degree if they apply without a M.S. degree. For non-English speaking applicants without a U.S. degree, a minimum TOEFL score of 523/70 is required.

* Paper/internet-based test score.
Master of Computer Science

30 credits.

This Professional Master’s degree program consists of 30 credit hours of coursework in computer science. This program is designed for those without a prior degree in computer science, or those who are primarily interested in a (non-thesis) program preparing them for careers as working computer science professionals in business and industry. A full-time student enrolled in the program should be able to complete the requirements in 1 to 1.5 years. Specializations in business, software engineering, networking and telecommunications and information systems are available. Admission requirements include:

Bachelor’s degree from an accredited university with a minimum cumulative GPA of at least 3.0/4.0. Combined verbal and quantitative GRE examination score of at least 900 and an analytic writing score of at least 2.5, for the post-October 2002 test. The GRE requirement is waived for students with a bachelor’s degree from an accredited college or university in the United States with a cumulative GPA of at least 3.0/4.0.

Prerequisites include knowledge of a high level programming language at the level of CS 201 (Java or C/C++ programming is required). Students with insufficient background in computer science will be required to take CS 401 and CS 402 (Introduction to Advanced Studies I and II) and to earn at least a B in these courses. These prerequisite courses do not count toward the 30 credit hour requirement.

A maximum of 12 credit hours of 400-level courses and a maximum of 6 credit hours of accelerated courses are allowed as part of the 30 credit hours requirement.

Master of Science in Computer Science/Master of Chemical Engineering

44 credit hours

This combined program in computer science and chemical engineering addresses the growing need for process engineers with expertise in computational modeling and simulation of chemical processes. Similarly, the program provides strong engineering background that is required today in many areas of computer science. The program is jointly offered by the Department of Computer Science and the Department of Chemical and Environmental Engineering. Students in this program earn both Master of Science in Computer Science and Master of Chemical Engineering degrees.

Students must fulfill the core course requirements of both departments. Students are required to take 18 credit hours in graduate chemical engineering courses (courses numbered 500 or higher) and 26 credit hours in computer science courses (of which 20 credit hours must be 500-level courses). The 18 credit hours in chemical engineering courses consist of 12 credits in core courses listed in the description of the Master of Chemical Engineering requirements and six credit hours from the following courses:

CHE 508 Process Design Optimization
CHE 536 Computational Techniques in Engineering
CHE 560 Statistical Quality and Process Control
Master of Computer Science with Specialization in Business

33 credit hours

This program is designed to help computer science professionals extend and deepen their technical and practical knowledge of the field while introducing themselves to core topics in modern business practices. To complete the program students must satisfy Master of Computer Science requirements and also take three specialization courses from the Stuart School of Business:

Specialization Courses

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Course Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>BUS 510</td>
<td>Building an Innovative &amp; Sustainable Business</td>
</tr>
<tr>
<td>BUS 550</td>
<td>Business Analytics for Competitive Advantage</td>
</tr>
<tr>
<td>MBA 501</td>
<td>Accounting for Strategic Decision Making</td>
</tr>
<tr>
<td>MBA 502</td>
<td>Emerging Issues in the Global Business Environment</td>
</tr>
<tr>
<td>MBA 504</td>
<td>Spreadsheet Modeling</td>
</tr>
<tr>
<td>MBA 506</td>
<td>Leading &amp; Managing Knowledge-Intensive Organizations</td>
</tr>
<tr>
<td>MBA 509</td>
<td>Financial Management in a Globalized World</td>
</tr>
<tr>
<td>MBA 511</td>
<td>Creating, Communicating, &amp; Delivering Customer Value</td>
</tr>
</tbody>
</table>

Note: Stuart School of Business tuition and fees apply to these courses. Applicants to the program are not required to take the GMAT.

Master of Computer Science with Specialization in Computational Intelligence

30 Credit Hours

This program is intended for students who are interested in ways in which computers may learn and adapt based on data so as to solve complex problems in various areas of computer science.

To qualify for the specialization in Computational Intelligence, students must satisfy general Master of Computer Science requirements and are also required to select our of the following specialization courses:

Specialization Courses

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Course Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS 480</td>
<td>Artificial Intelligence Planning &amp; Control</td>
</tr>
<tr>
<td>CS 512</td>
<td>Topics in Computer Vision</td>
</tr>
<tr>
<td>CS 522</td>
<td>Data Mining</td>
</tr>
<tr>
<td>CS 583</td>
<td>Probabilistic Graphical Models</td>
</tr>
<tr>
<td>CS 584</td>
<td>Machine Learning</td>
</tr>
<tr>
<td>CS 585</td>
<td>Natural Language Processing</td>
</tr>
</tbody>
</table>

With department approval, a course not on the list above may be substituted for one of the five specialization courses.

Master of Computer Science with Specialization in Cyber-Physical Systems

30 Credit Hours

This program is intended for students who are interested in learning how to work with embedded controllers with integrated sensors and networking abilities and utilize them for real-world applications.

To qualify for the specialization in Cyber-Physical systems, students must satisfy general Master of Computer Science requirements and are also required to select four of the following specialization courses:

Specialization Courses

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Course Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS 442</td>
<td>Mobile Applications Development</td>
</tr>
<tr>
<td>CS 552</td>
<td>Distributed Real-Time Systems</td>
</tr>
<tr>
<td>CS 553</td>
<td>Cloud Computing</td>
</tr>
<tr>
<td>CS 555</td>
<td>Analytic Models and Simulation of Computer Systems</td>
</tr>
<tr>
<td>CS 556</td>
<td>Cyber-Physical Systems: Languages and Systems</td>
</tr>
<tr>
<td>CS 557</td>
<td>Cyber-Physical Systems: Networking and Algorithms</td>
</tr>
</tbody>
</table>

IIT Graduate Bulletin 2014-2016 219
Computer Science

Master of Computer Science with Specialization in Data Analytics

30 Credit Hours

Intelligent analysis of large amounts of data is a crucial component in supporting business decisions. The Master of Science with Specialization in Data Analytics is intended for students interested in learning how to discover patterns in large amounts of data in information systems and how to use these to draw conclusions.

To qualify for the specialization in Data Analytics, students must satisfy general Master of Computer Science requirements and are also required to select four of the following specialization courses.

Specialization Courses
- CS 422 Data Mining
- CS 522 Data Mining
- CS 554 Data-Intensive Computing
- CS 583 Probabilistic Graphical Models
- CS 584 Machine Learning
- CS 585 Natural Language Processing

Master of Computer Science with Specialization in Database Systems

30 credit hours

This program is designed to provide in-depth knowledge of the principles of design and development of information systems. Students must satisfy Master of Computer Science requirements and are also required to select four specialization courses. Note that CS 522, CS 525, and CS 529 also count as Programming core courses.

Specialization Courses
- CS 425 Database Organization
- CS 520 Data Integration, Warehousing, and Provenance
- CS 521 Object-Oriented Analysis and Design
- CS 522 Data Mining
- CS 525 Advanced Database Organization
- CS 529 Information Retrieval
- CS 553 Cloud Computing
- CS 554 Data-Intensive Computing

Master of Computer Science with Specialization in Distributed and Cloud Computing

30 Credit Hours

The Master of Computer Science with a Specialization in Distributed and Cloud Computing is intended for students who are interested in learning about distributed systems and how they are applied to real-world problems, as well as how emerging cloud computing technologies can be used to implement some of the world's most popular services and applications.

To qualify for the specialization in Distributed and Cloud Computing, students must satisfy general Master of Computer Science requirements and are also required to select four of the following specialization courses. Below, CS 550 is also marked (Sys), because it also counts as a Systems core course.

Specialization Courses
- CS 546 Parallel and Distributed Processing
- CS 550 Advanced Operating Systems
- CS 552 Distributed Real-Time Systems
- CS 553 Cloud Computing
- CS 554 Data-Intensive Computing
- CS 570 Advanced Computer Architecture

Master of Computer Science with Specialization in Education

33 Credit Hours

The Master of Computer Science with a specialization in Education is designed to enable Computer Science students to further their technical education while opening a career path toward teaching computer science.

Courses for the MCS/Education degree program are taken from the Computer Science Department and the Department of Mathematics and Science Education (MSED). In addition to satisfying General MCS degree requirements, the program of study must include 24 credit hours of CS/CSP courses and the following 3 MSED courses, which are the first 3 required courses for a teaching certificate.

MSED Required Courses
- MSED 300 Instructional Methods/Strategies I
- MSED 500 Analysis of Classrooms II (Practicum and Seminar)
- MSED 554 Middle and Secondary Level Science Curriculum

OR
- MSED 555 Middle and Secondary Level Mathematics Curriculum
Master of Computer Science with Specialization in Finance

33 Credit Hours

The Master of Computer Science with a specialization in Finance is designed to enable Computer Science students to further their technical education while opening a path toward a career in finance.

Courses for the MCS/Finance degree program are taken from the Computer Science Department and the Department of Finance in the IIT Stuart School of Business. In addition to satisfying the General MCS Degree requirements, the program of study must include 24 credit hours of CS/CSP courses and the following 3 MSF courses (9 credit hours).

Required Finance Courses
- MSF 504 Valuation and Portfolio Management
- MSF 505 Futures, Options, and OTC Derivatives
- MSF 506 Financial Statement Analysis

Master of Computer Science with Specialization in Information Security and Assurance

30 Credit Hours

Information security, privacy, and information assurance are of prime importance in modern computer systems where data can be accessed from nearly everywhere. The Master of Computer Science with Specialization in Information Security and Assurance is intended for students interested in aspects of security and assurance in modern e-commerce applications.

To qualify for the specialization in Information Security and Assurance, students must satisfy general Master of Computer Science requirements and are also required to select four of the following specialization courses:

Specialization Courses
- CS 458 Information Security
- CS 525 Advanced Database Organization
- CS 549 Cryptography and Network Security
- CS 595 Topics in Computer Science
- CSP 544 System and Network Security

Master of Computer Science with Specialization in Networking and Communications

30 credit hours

This program is designed to provide an in-depth knowledge of the theories and practices in computer networking and telecommunications. Students must satisfy Master of Computer Science requirements and are also required to select four specialization courses. Note that CS 542, CS 544, and CS 547 also count as Systems core courses.

Specialization Courses
- CS 455 Data Communications
- CS 542 Computer Networks I: Fundamentals
- CS 544 Computer Networks II: Network Services
- CS 547 Wireless Networking
- CS 548 Broadband Networks
- CS 549 Cryptography and Network Security
- CS 555 Analytic Models and Simulation of Computer Systems
- CS 557 Cyber-Physical Systems: Networking and Algorithms

Master of Computer Science with Specialization in Software Engineering

30 credit hours

This program is designed to provide an in-depth knowledge of theory and practices in software engineering, including hands-on experience in software design, development and maintenance. Students must satisfy Master of Computer Science requirements and are also required to select four specialization courses. Note that CS 536 and CS 586 also count as Theory and Systems core courses respectively.

Specialization Courses
- CS 487 Software Engineering I
- CS 521 Object-Oriented Analysis and Design
- CS 536 Science of Programming
- CS 537 Software Metrics
- CS 586 Software Systems Architectures
- CS 587 Software Project Management
- CS 589 Software Testing and Analysis
Master of Science in Computer Science

32 credit hours

The purpose of this program is to prepare students for the Ph.D. program and/or a research/development career in the industry in the field of computer science. Students have the option to pursue thesis research or project under the guidance of a faculty advisor.

Program Requirements

All programs require a core curriculum of 12 credit hours and 20 credit hours of elective courses, which may include a thesis or project. If a thesis or project is included in the program, the student, with a faculty advisor, develops a program of study that specifies the supportive and elective program and describes the thesis or project. The program of study must consist of at least 32 credit hours, at least 20 of which must be 500-level computer science courses. Up to six credits of accelerated courses may be applied to the degree. (Students should see the definition of “accelerated courses” below.)

A student may choose from three options to complete the degree:

Option 1: Master’s thesis: Coursework and up to five hours of CS 591 for a total of 32 hours. The result is a master’s thesis.

Option 2: Master’s project: coursework and up to five hours of CS 597 for a total of 32 hours. The result is a project that results in one of the following:

1. A high-quality paper submitted for publication as an article or as a technical report.
2. A high-quality piece of software. The software should be of distribution quality, but can be proprietary.

Option 3: 32 credit hours of coursework. A student must complete 32 hours of regular coursework including electives and core courses with a GPA of 3.0/4.0 or better.

Students are required to take courses in three core areas: Programming, Systems, and Theory. The student is required to take at least one course from the Programming area, at least one course from the Systems area, and at least two courses from the Theory area. The list below contains the core course offerings in the M.S. program:

Programming Core Courses

CS 511 Topics in Computer Graphics
CS 512 Topics in Computer Vision
CS 525 Advanced Database Organization
CS 540 Syntactic Analysis of Programming Languages
CS 541 Topics in Compiler Construction
CS 546 Parallel and Distributed Processing
CS 551 Operating System Design and Implementation
CS 553 Cloud Computing

Systems Core Courses

CS 542 Computer Networks I: Fundamentals
CS 544 Computer Networks II: Network Services
CS 547 Wireless Networking
CS 550 Advanced Operating Systems
CS 555 Analytic Models and Simulation of Computer Systems
CS 570 Advanced Computer Architecture
CS 586 Software Systems Architectures

Theory Core Courses

CS 530 Theory of Computation
CS 533 Computational Geometry
CS 535 Design and Analysis of Algorithms
CS 536 Science of Programming
CS 538 Combinatorial Optimization
CS 539 Game Theory: Algorithms and Applications
Master of Data Science
Collaborative Program with the Department of Applied Mathematics

33 credit hours

This Professional Master’s degree program consists of 33 credit hours of coursework, including a practicum, in data science. The program is designed primarily for those with previous degrees or experience in computer science, statistics, mathematics, natural sciences, or business, who are interested in preparing for a career as a data science professional in business and industry. Enrolled full-time, the program can be completed in a year, including one summer.

Admission Requirements

A Bachelor’s degree from an accredited university with a minimum cumulative GPA of at least 3.0/4.0. Combined verbal and quantitative GRE examination score of at least 304 and an analytic writing score of at least 3.0, for the post-October 2002 test. The GRE requirement is waived for students with a bachelor’s degree from an accredited college or university in the United States with a cumulative GPA of at least 3.0/4.0.

Prerequisites include knowledge of a high level programming language at the level of CS 201 (Java or C/C++-programming is required), a data structures course at the level of CS 331, experience with database programming at the level of CS 425, linear algebra at the level of MATH 332, and probability and statistics at the level of MATH 474. Information on these courses is available in this catalog.

Students with insufficient background in computer science and/or mathematics will be required to take the relevant prerequisite courses and earn at least a B grade in each. These prerequisite courses do not count toward the 33 credit hour requirement.

Program Requirements

Coursework includes 18 credit hours of required core courses and 6 credit hours of CSP/MATH 572 Data Science Practicum. At least 9 credit hours must be taken of 500-level CS or CSP courses and 9 credit hours of 500-level MATH courses, not including the CSP/MATH 572 Data Science Practicum. Students must also take one semester of CSP/MATH 570 Data Science Seminar.

Data Science Core Courses
- COM 523 Communicating Science
- CS 587 Software Project Management
- CS 584 Machine Learning
- OR
- MATH 509 Statistical Learning
- OR
- CS 525 Advanced Database Organization
- OR
- CS 554 Data-Intensive Computing
- MATH 564 Applied Statistics
- MATH 571 Data Preparation and Analysis

Data Science Electives

Computational Fundamentals
- CS 425 Database Organization
- CS 450 Operating System
- CS 535 Design and Analysis of Algorithms
- CS 546 Parallel and Distributed Processing
- CS 553 Cloud Computing
- CS 589 Software Testing and Analysis

Computer Science Applications
- CS 512 Topics in Computer Science
- CS 529 Information Retrieval
- CS 556 Cyber-Physical Systems: Languages and Systems
- CS 557 Cyber-Physical Systems: Networking and Algorithms
- CS 583 Probabilistic Graphical Models
- CS 585 Natural Language Processing

Mathematics, Probability, and Statistics
- MATH 532 Linear Algebra
- MATH 540 Probability
- MATH 542 Stochastic Processes
- MATH 565 Monte Carlo Methods in Finance
- MATH 567 Advanced Design of Experiments
- MATH 574 Bayesian Computational Statistics

Mathematical and Scientific Computing
- MATH 577 Computational Mathematics I
- MATH 578 Computational Mathematics II
- MATH 590 Meshfree Methods
- BIOL 550 Bioinformatics
- PHYS 440 Computational Physics
Master of Telecommunications and Software Engineering (M.T.S.E.)
Collaborative Program with the Department of Electrical and Computer Engineering

30 credit hours

The Master of Telecommunications and Software Engineering (M.T.S.E.) is a course-only degree program that prepares students for professional practice in telecommunications and information technologies. The program, jointly offered by the Electrical and Computer Engineering (ECE) and Computer Science (CS) departments, can be completed in one year of full-time study.

Admission Requirements

A person holding a B.S.E.E., B.S.C.P.E. or B.S.C.S. degree has the necessary broad background to undertake the M.T.S.E. program. A student without adequate background in specific areas is required to demonstrate proficiency in prerequisite courses; an abbreviated list is given below.

Specific proficiency courses will be detailed for each student at the time of admission to the M.T.S.E. program. Proficiency in a course may be demonstrated by completing the course with a grade of A or B, or by achieving a grade of A or B in a proficiency examination administered by the ECE or the CS department. Students should contact the departmental advisor for more details on prerequisites and proficiency requirements.

Students interested in the M.T.S.E. degree whose B.S. degree is not in electrical engineering, computer engineering, or computer science should contact the departmental advisor before applying.

Computer Science Prerequisites

CS 201 Accelerated Introduction to Computer Science
CS 401 Introduction to Advanced Studies

Electrical and Computer Engineering Prerequisites

ECE 211 Circuit Analysis I
ECE 213 Circuit Analysis II
ECE 308 Signals and Systems
MATH 252 Introduction of Differential Equations

Program Requirement

The M.T.S.E. is a professional master’s degree requiring a minimum of 30 credit hours of advisor-approved coursework. The M.T.S.E program of studies must include a minimum of 12 credit hours of ECE coursework and a minimum of 12 credit hours of CS coursework. Four required courses and one elective course from each of the three categories given below must appear on the M.T.S.E. program of studies.

Required Courses

CS 586 Software Systems Architecture
CS 587 Software Project Management
ECE 513 Communication Engineering Fundamentals
ECE 541 Performance Evaluation of Computer Networks (can be substituted with ECE 542)

Elective Categories

I. Software Engineering

CS 521 Object-Oriented Analysis and Design
CS 537 Software Metrics
CS 589 Software Testing and Analysis

II. Telecommunication Systems

CS 544 Computer Networks II: Network Services
CS 548 Broadband Networks
CS 555 Analytic Models and Simulation of Computer Systems
ECE 545 Advanced Computer Networks

III. Telecommunications

ECE 504 Wireless Communication System Design
ECE 515 Modern Digital Communications
ECE 519 Coding for Reliable Communications

The remaining nine credits of coursework may be taken from courses listed above, or other courses approved by the M.T.S.E. advisor. Students with no background in communications or software engineering should consider including in their programs of study:

CS 450 Operating Systems
CS 455 Data Communications
CS 487 Software Engineering I
ECE 403 Digital and Data Communication Systems

Other courses that students in this program typically choose from include:

ECE 437 Digital Signal Processing I
ECE 511 Analysis of Random Signals
ECE 514 Digital Communication Principles
CS 542 Computer Networks I: Fundamentals
CS 588 Advanced Software Engineering Development

With advisor approval, the M.T.S.E. program can include up to two credit hours of Master’s Seminar (ECE 595, ECE 596). It can also include up to four credits of accelerated courses.
**Doctor of Philosophy**

85 credit hours if without M.S. degree; 54 credit hours if with M.S. degree.

Qualifying exam
Comprehensive exam
Dissertation and oral defense

The Ph.D. is awarded in recognition of a significant original contribution to one of the fields of computer science and a high level of mastery in several fields of computer science and a significant original contribution to one of those fields. Students work with faculty members to develop programs to match individual interests. The goal is to develop computer scientists who can take complex, undefined problems and restructure and resolve them through imaginative application of their knowledge. Graduates typically go on to teaching and/or research positions in industry and universities. The degree normally requires three to four years beyond the master’s degree for full-time students. Part-time students can also enter the program but will need more time to complete the degree. Generally, students can enter the program with either a B.S. degree or an M.S. degree in related fields. The requirements of the Ph.D. program are described separately as follows.

---

**Requirements for Students Entering With a B.S. Degree**

85 credit hours
Qualifying exam
Comprehensive exam
Dissertation and oral defense

**Admission Requirements**

The Ph.D. (post B.S.) program (called the direct Ph.D. program) encourages bright and highly motivated students to participate in a research program immediately after the B.S. degree.

The applicants should have a B.S degree in computer science. Admission to the program is competitive and depends on a student’s GRE score and it is expected that applicants will have a high grade-point average. For non-English speaking applicants without a U.S. degree, a minimum TOEFL score of 550 is required. If the TOEFL score is less than 600, the applicant is required to take the English Proficiency Exam administered by the IIT Humanities Department. Applicants must submit three letters of recommendation.

---

**Requirements for Students Entering With an M.S. Degree**

54 credit hours
Qualifying exam
Comprehensive exam
Dissertation and oral defense

**Admission Requirements**

The applicants should have an M.S degree in computer science or related fields. Admission to the program is competitive and depends on a student’s GRE score and it is expected that applicants will have a high grade-point average. For non-English speaking applicants without a U.S. degree, a minimum score of 70 on the internet based TOEFL or 523 on the paper based TOEFL; or 47 on the PTE; or 5.5 on the IELTS is required for admission consideration. Applicants must submit three letters of recommendation.
Doctor of Philosophy - continued

Program Requirements (for students with a B.S. degree)

The program requires students to complete at least 85 and at most 128 advisor-approved semester credit hours of study. This must include

- 0-12 credits of 400-level courses
- 36-54 credits of 500- and 600-level courses. Among them, at most 6 credits come from outside the Computer Science Department of IIT. Credits from CS 595 are allowed.
- 6-12 credits of CS 597 (Reading and Special Problems)
- 24-48 credits of CS 691 (Research/Thesis for Ph.D. degree). Note CS 691 can only be taken after passing the Ph.D. qualifying exam.
- 1 credit of CS 695 Ph.D. seminar

Notice that no credits will be given to accelerated courses (700-level courses). No credits are given to courses in which the student earns a grade of C or below. The student may have to take some other courses as required by the advisor. The credit hours of 500-level coursework must include five core courses with at least one each from Groups I-III described below and at least one course from each of two different groups from among Groups IV-VII. Advanced courses may be substituted after approval of the department. The groups of core courses are:

Group I: Theory of Computation
CS 530 Theory of Computation
CS 533 Computational Geometry
CS 535 Design and Analysis of Algorithms
CS 538 Combinatorial Optimization
CS 539 Game Theory: Algorithms and Applications

Group II: Systems
CS 546 Parallel and Distributed Processing
CS 550 Advanced Operating Systems
CS 570 Advanced Computer Architecture

Group III: Programming Languages
CS 536 Science of Programming
CS 540 Syntactic Analysis of Programming Languages
CS 541 Topics in Compiler Construction
CS 545 Distributed Computing Landscape

Group IV: Networks
CS 542 Computer Networks I: Fundamentals
CS 544 Computer Networks II: Network Services

Group V: Databases
CS 525 Advanced Database Organization

Group VI: Software Engineering
CS 586 Software Systems Architectures

Group VII: Computational Intelligence
CS 512 Topics in Computer Vision
CS 583 Probabilistic Graphical Models
CS 584 Machine Learning
CS 585 Natural Language Processing

M.S. Exit from Program

Students wishing to leave the direct Ph.D. program with the degree of Master of Science in Computer Science must satisfy all the requirements of the Master’s degree and either write an M.S. thesis or pass the Ph.D. qualifying examination.

Ph.D. Qualifying Examination

The Ph.D. Qualifying Exam has two parts: a written examination and an oral examination. The written examination is used to judge a student’s breadth of knowledge; the oral examination is used to judge a student’s research potential. The first attempt in oral examination and the written examination must be taken no later than a student’s 5th semester. The second attempt must be taken no later than a student’s 6th semester. These requirements hold for both full-time and part-time students. The written examination is divided into three, independent “area” examinations. To pass the written examination, a student must pass all the area examinations.

Area examinations can be taken in the same or different semesters. A student who fails an area examination can retake the area examination, but only once. Passing a relevant core course with “A” when registered in the PhD section of hat course qualifies as passing the respective area examination. See the computer science webpage for more detail of qualifying examinations.

Comprehensive (Research Proposal) Examination

The purpose of the Comprehensive Examination is to ensure that the candidate has the background to carry out successful research in the chosen area and that the research problem is properly formulated and has sufficient scholarly merit. The student (in concert with the student’s research advisor) must develop a written research proposal containing a literature review, a proposed research topic, and a program of research based upon this topic, and then present the proposal orally as well.

The student must request appointment of an examination committee using Form 301. The examination committee may consist of from four to seven members. It must include at least three full-time faculty members from the Computer Science Department and one full-time faculty member from another department in the university. Other committee members from inside or outside the university may be chosen. The student should consult with his/her research advisor concerning the makeup of the committee.

Thesis Defense

Each student must present an oral defense of his/her Ph.D. Thesis. The Thesis Review Committee is appointed in much the same way as the Ph.D. Comprehensive Examination Committee. It will examine the written thesis and examine the student during the oral defense. All Ph.D. Thesis Defenses are open to the public.
Doctor of Philosophy - continued

Program Requirements (for students with an M.S. degree)

If the student has an M.S. degree in computer science, the program requires the student to complete at least 54 advisor-approved semester credit hours of study. This must include:

- 0-12 credits of 400-level courses
- 18-30 credits of 500- and 600-level courses. Amongst them, at most 6 credits come from outside the computer science department. Credits from CS 595 are allowed.
- 3-12 credits of CS 597 (Reading and Special Problems)
- 24-48 credits of CS 691 (Research/Thesis for Ph.D. degree). Note CS 691 can only be taken after passing the Ph.D. qualifying exam.
- 1 credit of CS 695 Ph.D. seminar

Notice that no credits will be given to accelerated courses (700-level courses). No credits are given to courses in which the student earns a grade of C or below. The student may have to take some other courses as required by the advisor. The credit hours of 500-level coursework must include three core courses with at least one each from Groups I-III described before. Advanced courses may be substituted after approval of the department.

If the student has an M.S. degree in a field other than computer science, the program requires the student to complete at least 60 advisor-approved semester credit hours of study. This must include:

- 0-12 credits of 400-level courses
- 24-30 credits of 500- and 600-level courses. Among them, at most 6 credits may come from outside the computer science department. Credits from CS 595 are allowed.
- 3-12 credits of CS 597 (Reading and Special Problems)
- 24-48 credits of CS 691 (Research/Thesis for Ph.D. degree). Note CS 691 can only be taken after passing the Ph.D. qualifying exam.
- 1 credit for CS 695 (the Ph.D. seminar)

Notice that no credits will be given to accelerated courses (700-level courses). No credits are given to courses in which the student earns a grade of C or below. The student may have to take some other courses as required by the advisor. The credit hours of 500-level coursework must include five core courses with at least one each from Groups I-III described before and at least two courses from two different groups from among Groups IV-VII. Advanced courses may be substituted after approval of the department.

Ph.D. Qualifying Examination

The Ph.D. Qualifying Examination has two parts: a written examination and an oral examination. The written exam is used to judge a student’s breadth of knowledge; the oral exam is used to judge a student’s research potential. The first attempt at the oral examination and the written examination must be made no later than a student’s 3rd semester. The second attempt must be made no later than a student’s 4th semester. These requirements hold for both full-time and part-time students.

The written examination is divided into three, independent “area” examinations. To pass the written examination, a student must pass all the area examinations. Area examinations can be taken in the same or different semesters. A student who fails an area examination can retake the area examination, but only once. Passing a relevant core course with “A” when registered in the PhD section of that course qualifies as passing the respective area examination. See the computer science web page for more details about the qualifying examinations.

Comprehensive (Research Proposal) Examination

The purpose of the Comprehensive Examination is to ensure that the candidate has the background to carry out successful research in the chosen area and that the research problem is properly formulated and has sufficient scholarly merit. The student (in concert with the student’s research advisor) must develop a written research proposal containing a literature review, a proposed research topic, and a program of research based upon this topic, and then present it orally as well.

The student must request appointment of an examination committee on Form 301. The examination committee may consist of from four to seven members. It must include at least three full-time faculty members from the Computer Science Department and one full-time faculty member from another department in the university. Other committee members from inside or outside the university may be chosen. The student should consult with his/her research advisor concerning the makeup of the committee.

Thesis Defense

Each student must present an oral defense of his/her Ph.D. Thesis. The Thesis Review Committee is appointed in much the same way as the Ph.D. Comprehensive Examination Committee. It will examine the written thesis and examine the student during the oral defense. All Ph.D. Thesis Defenses are open to the public.
Certificate Programs
Certificate programs offer working professionals an opportunity to increase their knowledge and skills in the specific areas of computer science. A certificate representing proven academic performance is presented after the required coursework is completed with a GPA of 3.0/4.0. Courses taken may be later applied toward an M.S. degree program. Applicants should have a bachelors degree from an accredited college or university. The degree need not be in computer science.

Computational Intelligence Certificate
Nine hours from the following:
- CS 480  Artificial Intelligence Planning & Control
- CS 512  Topics in Computer Vision
- CS 522  Data Mining
- CS 583  Probabilistic Graphical Models
- CS 584  Machine Learning
- CS 585  Natural Language Processing

Cyber-Physical Systems Certificate
Nine hours from the following:
- CS 442  Mobile Applications Development
- CS 552  Distributed Real-Time Systems
- CS 553  Cloud Computing
- CS 555  Analytic Models and Simulation of Computer Systems
- CS 556  Cyber-Physical Systems: Languages and Systems
- CS 557  Cyber-Physical Systems: Networking and Algorithms

Data Analytics Certificate
Nine hours from the following:
- CS 422  Data Mining
- CS 522  Data Mining
- CS 554  Data-Intensive Computing
- CS 583  Probabilistic Graphical Models
- CS 584  Machine Learning
- CS 585  Natural Language Processing

Database Systems Certificate
Nine hours from the following:
- CS 425  Database Organization
- CS 520  Data Integration, Warehousing, & Provenance
- CS 521  Object-Oriented Analysis and Design
- CS 522  Data Mining
- CS 525  Advanced Database Organization
- CS 529  Information Retrieval
- CS 553  Cloud Computing
- CS 554  Data-Intensive Computing

Distributed and Cloud Computing Certificate
Nine hours from the following:
- CS 451  Introduction to Parallel & Distributed Computing
- CS 546  Parallel and Distributed Processing
- CS 550  Advanced Operating Systems
- CS 552  Distributed Real-Time Systems
- CS 553  Cloud Computing
- CS 554  Data-Intensive Computing
- CS 570  Advanced Computer Architecture
Information Security and Assurance Certificate

Nine hours from the following:

<table>
<thead>
<tr>
<th>Course</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS 458</td>
<td>Information Security</td>
</tr>
<tr>
<td>CS 525</td>
<td>Advanced Database Organization</td>
</tr>
<tr>
<td>CS 549</td>
<td>Cryptography and Network Security</td>
</tr>
<tr>
<td>CS 595</td>
<td>Topics in Computer Science: Information</td>
</tr>
<tr>
<td></td>
<td>Security, Privacy, and Assurance</td>
</tr>
<tr>
<td>CSP 544</td>
<td>System and Network Security</td>
</tr>
</tbody>
</table>

Networking and Communications Certificate

Nine hours from the following:

<table>
<thead>
<tr>
<th>Course</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS 455</td>
<td>Data Communications</td>
</tr>
<tr>
<td>CS 542</td>
<td>Computer Networks I: Fundamentals</td>
</tr>
<tr>
<td>CS 544</td>
<td>Computer Networks II: Network Services</td>
</tr>
<tr>
<td>CS 547</td>
<td>Wireless Networking</td>
</tr>
<tr>
<td>CS 548</td>
<td>Broadband Networks</td>
</tr>
<tr>
<td>CS 549</td>
<td>Cryptography and Network Security</td>
</tr>
<tr>
<td>CS 555</td>
<td>Analytic Models &amp; Simulation of Computer</td>
</tr>
<tr>
<td></td>
<td>Systems</td>
</tr>
</tbody>
</table>

Software Engineering Certificate

Nine hours from the following:

<table>
<thead>
<tr>
<th>Course</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS 487</td>
<td>Software Engineering I</td>
</tr>
<tr>
<td>CS 521</td>
<td>Object-Oriented Analysis and Design</td>
</tr>
<tr>
<td>CS 536</td>
<td>Science of Programming</td>
</tr>
<tr>
<td>CS 537</td>
<td>Software Metrics</td>
</tr>
<tr>
<td>CS 586</td>
<td>Software Systems Architecture</td>
</tr>
<tr>
<td>CS 587</td>
<td>Software Project Management</td>
</tr>
<tr>
<td>CS 589</td>
<td>Software Testing and Analysis</td>
</tr>
</tbody>
</table>

Accelerated Programs

The department offers accelerated courses for credit in several areas of computer science. These courses go beyond traditional core topics and are designed for working professionals who are interested in keeping abreast of rapidly changing technologies. Accelerated courses provide an opportunity for degree-seeking students at IIT to complete M.S. degree requirements in a shorter time period. If taken by non-degree students, these courses can be applied towards requirements for an M.S. degree at IIT.
Course Descriptions

CS 511  
Topics in Computer Graphics  
Covers advanced topics in computer graphics. The exact course contents may change based on recent advances in the area and the instructor teaching it. Possible topics include: Geometric modeling, Subdivision surfaces, Procedural modeling, Warping and morphing, Model reconstruction, Image based rendering, Lighting and appearance, Texturing, Natural phenomena, Nonphotorealistic rendering Particle systems, Character animation, Physically based modeling and animation.  
Prerequisite(s): [(CS 411)]  
(3-0-3)

CS 512  
Topics in Fundamental Vision  
Introduction to fundamental topics in computer vision and the application of statistical estimation techniques to this area. Intended to give the student a good basis for work in this important field. Topics include: Feature extraction, Probabilistic modeling, Camera calibration, Epipolar geometry, Statistical estimation, Model reconstruction, Statistical filtering, Motion estimation, Recognition, Shape from single image cues.  
Prerequisite(s): [(CS 430)]  
(3-0-3)

CS 513  
Geospatial Vision & Visualization  
Geospatial information has become ubiquitous in everyday life as evidenced by on-line mapping services such as NOKIA Here Map, Microsoft Bing Map, the “place” features on social network websites such as Facebook, and navigation apps on smart phones. Behind the scenes is digital map content engineering that enables all types of location-based services. Course material will be drawn from the instructor’s research and development experience at NOKIA Location and Commerce (formerly NAVTEQ), the Chicago-based leading global provider of digital map, traffic, and location data. This course will provide a comprehensive treatment of computer vision, image processing and visualization techniques in the context of digital mapping, global positioning and sensing, next generation map making, and three-dimensional map content creations. Real world problems and data and on-site industry visits will comprise part of the course curriculum.  
(3-0-3)

CS 520  
Data Integration, Warehousing, & Provenance  
This course introduces the basic concepts of data integration, data warehousing, and provenance. We will learn how to resolve structural heterogeneity through schema matching and mapping. The course introduces techniques for querying several heterogeneous data sources at once (data integration) and translating data between databases with different data representations (data exchange). Furthermore, we will cover the data-warehouse paradigm including the Extract-Transform-Load (ETL) process, the data cube model and its relational representations (such as snowflake and star schema), and efficient processing of analytical queries. This will be contrasted with Big Data analytics approaches that (besides other differences) significantly reduce the upfront cost of analytics. When feeding data through complex processing pipelines such as data exchange transformations or ETL workflows, it is easy to lose track of the origin of data. Therefore, in the last part of the course we cover techniques for representing and keeping track of the origin and creation process of data (its provenance).  
The course emphasizes practical skills through a series of homework assignments that help students develop a strong background in data integration systems and techniques. At the same time, it also addresses the underlying formalisms. For example, we will discuss the logic based languages used for schema mapping and the dimensional data model as well as their practical application (e.g., developing an ETL workflow with rapid miner and creating a mapping between two example schemata). The literature reviews will familiarize students with data integration and provenance research.  
Prerequisite(s): [(CS 425)]  
(3-0-3)

CS 521  
Object-Oriented Analysis & Design  
This course describes a methodology that covers a wide range of software engineering techniques used in system analysis, modeling, and design. These techniques integrate well with software process management techniques and provide a framework for software engineers to collaborate in the design and development process. The methodology features the integration of concepts, including software reusability, frame works, design patterns, software architecture, software component design, use-case analysis, event-flow analysis, event-message analysis, behavioral-life cycle analysis, feature, multiple-product, risk and rule analysis, and automatic code generation. (Credit will not be given for CS 521 if CST51 is taken)  
Prerequisite(s): [(CS 445) OR (CS 487)]  
(3-0-3)

CS 522  
Data Mining  
Continued exploration of data mining algorithms. More sophisticated algorithms such as support vector machines will be studied in detail. Students will continuously study new contributions to the field. A large project will be required that encourages students to push the limits of existing data mining techniques.  
Prerequisite(s): [(CS 422)]  
(3-0-3)
CS 525  
Advanced Database Organization  
Comprehensive coverage of the problems involved in database system implementation and an in-depth examination of contemporary structures and techniques used in modern database management systems. Teaches advanced skills appropriate for DBMS architects and designers, database specialists, and the designers and developers of client/server and distributed systems. Focus is on transaction management, database structures and distributed processing.  
Prerequisite(s): [(CS 425)]  
(3-0-3)

CS 529  
Information Retrieval  
The course covers the advanced topics in Information Retrieval. The topics such as Summarization, cross-lingual, Meta-Search, Question Answering, Parallel and distributed IR systems are discussed. The students get involved in research ideas, and get involved in individual and group projects.  
Prerequisite(s): [(CS 429)]  
(3-0-3)

CS 530  
Theory of Computation  
Computability topics such as Turing machines, nondeterministic machines, undecidability, and reducibility. Computational complexity topics such as time complexity, NP-completeness and intractability, time and space hierarchy theorems. Introduces the complexity classes P, NP, NL, L, PSPACE, NC, NC1, BPP and their complete problems.  
Prerequisite(s): [(CS 430)]  
(3-0-3)

CS 531  
Topics in Automata Theory  
Topics selected from mathematical systems and automata theory, decision problems, realization and minimization, algebraic decomposition theory and machines in a category.  
Prerequisite(s): [(CS 430)]  
(3-0-3)

CS 533  
Computational Geometry  
This course covers fundamental algorithms and data structures for convex hulls, Voronoi diagrams, Delaunay triangulation, Euclidean spanning trees, point location, and range searching. Also included are lower bounds and discrepancy theory. Optimization in geometry will be covered. This includes fixed dimensional linear programming and shortest paths. Graphical data structures such as BSP trees will be covered.  
Prerequisite(s): [(CS 430)]  
(3-0-3)

CS 535  
Design & Analysis of Algorithms  
Design of efficient algorithms for a variety of problems, with mathematical proof of correctness and analysis of time and space requirements. Topics include lower bounds for sorting and medians, amortized analysis of advanced data structures, graph algorithms (strongly connected components, shortest paths, minimum spanning trees, maximum flows and bipartite matching) and NP-Completeness.  
Prerequisite(s): [(CS 430)]  
(3-0-3)

CS 536  
Science of Programming  
Formal specification of how programs execute operational semantics, and how mathematical functions programs compute denotational semantics, and how to use logic to characterize properties and invariants of the program execution (axiomatic semantics).  
Prerequisite(s): [(CS 331) OR (CS 401)]  
(3-0-3)

CS 537  
Software Metrics  
Prerequisite(s): [(CS 487)]  
(3-0-3)

CS 538  
Combinatorial Optimization  
Linear programs and their properties. Efficient algorithms for linear programming. Network flows, mininum cost flows, maximum matching, weighted matching, matroids.  
Prerequisite(s): [(CS 430) and a linear algebra course]  
(3-0-3)

CS 539  
Game Theory: Algorithms & Applications  
This course focuses on computational issues in the theory of games, economics, and network design. Interest in the algorithmic aspects of games is motivated by the computational issues of fundamental aspects of games and economic theory, e.g. Nash equilibrium and market equilibrium. Computing and approximating Nash equilibrium will be studied. Of considerable interest to the computer science community are problems that arise from the Internet and computer networks and are similar to issues that arise in traditional transport networks, e.g. Wardrop equilibrium.  
Prerequisite(s): [(CS 430) OR (CS 530)]  
(3-0-3)

CS 540  
Syntactic Analysis of Programming Languages  
Formal definition of syntax with emphasis on context-free languages. Elementary techniques for scanning and parsing programming languages. Symbol table management. Semantic routines and code generation. The class will write a simple translator.  
Prerequisite(s): [(CS 440)]  
(3-0-3)

CS 541  
Topics in Compiler Construction  
Advanced topics in compiler construction, including incremental and interactive compiling, error correction, code optimization, models of code generators, etc. The objective of the course is to provide an in-depth coverage of compiler optimization techniques, including both classical optimization and areas of current interest in compiler research.  
Prerequisite(s): [(CS 440)]  
(3-0-3)
CS 542  
**Computer Networks I Fundamentals**  
This course focuses on the engineering and analysis of network protocols and architecture in terms of the Internet. Topics include content distribution, peer-to-peer networking, congestion control, unicast and multicast routing, router design, mobility, multimedia networking quality of service, security and policy-based networking.  
Prerequisite(s): [(CS 455)](3-0-3)

CS 544  
**Computer Networks II: Network Services**  
Qualitative and quantitative analysis of networks. A combination of analytical and experimental analysis techniques will be used to study topics such as protocol delay, end-to-end network response time, intranet models, Internet traffic models, web services availability, and network management.  
Prerequisite(s): [(CS 542) OR (ECE 545)](3-0-3)

CS 545  
**Distributed Computing Landscape**  
Introduction to the theory of concurrent programming languages. Topics include formal models of concurrent computation such as process algebras, nets, and actors; high-level concurrent programming languages and their operational semantics; and methods for reasoning about correctness and complexity of concurrent programs.  
Prerequisite(s): [(CS 450)](3-0-3)

CS 546  
**Parallel & Distributed Processing**  
This course covers general issues of parallel and distributed processing from a user's point of view which includes system architectures, programming, performance evaluation, applications, and the influence of communication and parallelism on algorithm design.  
Prerequisite(s): [(CS 430 and CS 450)](3-0-3)

CS 547  
**Wireless Networking**  
This course introduces cellular/PCS systems, short-range mobile wireless systems, fixed wireless systems, satellites, and ad hoc wireless systems. It explains in detail the underlying technology as well as regulations, politics, and business of these wireless communications systems. It looks beyond the hype, examining just what is and is not possible with present-day and future wireless systems. As an advanced graduate course, it will combine extensive reading and in-class discussion of the research literature with in-depth independent research projects of students' own choosing.  
Prerequisite(s): [(CS 455)](3-0-3)

CS 548  
**Broadband Networks**  
The course studies the architectures, interfaces, protocols, technologies, products and services for broadband (high-speed) multimedia networks. The key principles of the protocols and technologies used for representative network elements and types of broadband network are studied. Specifically, cable modems, Digital Subscriber Lines, Power Lines, wireless 802.16 (WiMax), and broadband cellular Internet are covered for broadband access; for broadband Local Area Networks (LANs), Gigabit Ethernet, Virtual LANs and wireless LANs (802.11 WiFi and Bluetooth) are discussed; for broadband Wide Area Networks (WANs) the topics covered include optical networks (SONET/SDH,DWDM, optical network nodes, optical network nodes, optical switching technologies), frame-relay, ATM, wire-speed routers, IP switching, and MPLS. Also, quality of service issues in broadband networks and a view of the convergence of technologies in broadband networks are covered.  
Prerequisite(s): [(CS 455)](3-0-3)

CS 549  
**Cryptography & Network Security**  
This course provides an introduction to the theory and practice of cryptography and network security. The course covers conventional encryption such as classical encryption techniques, modern encryption techniques and encryption algorithms. Students are introduced to the basic number theory, which is used as the foundation for public-key encryption. The public-key cryptography such as encryption methods and digital signatures is covered. Message authentication and hash functions are also discussed. Students will learn techniques of key management, secret sharing and conducting interactive proofs. In addition, the practical network and security protocols are discussed.  
Prerequisite(s): [(CS 430)](3-0-3)

CS 550  
**Advanced Operating Systems**  
Advanced operating system design concepts such as interprocess communication, distributed processing, replication and consistency, fault tolerance, synchronization, file systems. Study of systems highlighting these concepts.  
Prerequisite(s): [(CS 450)](3-0-3)

CS 551  
**Operating System Design & Implementation**  
This course covers in detail the design and implementation of processes, interprocess communication, semaphores, monitors, message passing, scheduling algorithm, input/output, device drivers, memory management, file system design, security and protection mechanisms. The hardware-software interface and the user process-system call-kernel interface are examined in detail. Students modify and extend a multiuser operating system.  
Prerequisite(s): [(CS 450)](3-0-3)
CS 552
Distributed Real-Time Systems
With the advancement of computer hardware, embedded devices, and network technology, real-time applications have become pervasive, ranging from smart automobiles to automated traffic control. Different from general-purpose applications, correct executions of real-time applications depend on both functional correctness and temporal correctness. This course is to study the fundamentals of distributed real-time computing with the focus on its temporal aspects.
Prerequisite(s): [(CS 450)]
(3-0-3)

CS 553
Cloud Computing
This course is a tour through various topics and technologies related to cloud computing. Students will explore solutions and learn design principles for building large network-based systems to support both compute-intensive and data-intensive applications across geographically distributed infrastructure. Topics include resource management, programming models, application models, system characterizations, and implementations. Discussions will often be grounded in the context of deployed cloud computing systems such as Amazon EC2 and S3, Microsoft Azure, Google App Engine, Eucalyptus, Nimbus, OpenStack, Google's MapReduce, Yahoo's Hadoop, Microsoft's Dryad, Sphere/Sector, and many other systems.
The course involves lectures, outside invited speakers, discussions of research papers, programming assignments, and a major project (including both a written report and an oral presentation).
Prerequisite(s): [(CS 450) OR (CS 455)]
(3-0-3)

CS 554
Data-Intensive Computing
This course is a tour through various research topics in distributed data-intensive computing, covering topics in cluster computing, grid computing, supercomputing, and cloud computing. The course will explore solutions and learn design principles for building large network-based computational systems to support data-intensive computing. This course is geared for junior/senior-level undergraduates and graduate students in computer science.
Prerequisite(s): [(CS 450)]
(3-0-3)

CS 555
Analytic Models & Simulation of Computer Systems
Analytic and simulation techniques for the performance analysis of computer architecture, operating systems and communication networks. Rigorous development of queuing models. Study of simulation languages and models.
Prerequisite(s): [(CS 450)]
(3-0-3)

CS 556
Cyber-Physical Systems: Languages & Systems
Different from general-purpose and traditional computer applications, cyber-physical systems have both continuous and discrete components, hence requiring new methodologies to integrate traditional continuous control theory/systems with traditional discrete software systems. The focus of this course is to discuss and understand the challenges in emerging cyber-physical systems and to explore possible solutions from the perspectives of systems specification, system modeling, programming languages, systems designs, and software engineering. This course will focus on the languages and systems aspects of cyber-physical systems.
(3-0-3)
CS 579
Online Social Network Analysis
This course will explore the latest algorithms for analyzing online social networks, considering both their structure and content. Fundamentals of social graph theory will be covered including distance, search, influence, community discovery, diffusion, and graph dynamics. Fundamentals of text analysis will also be covered with an emphasis on the type of text used in online social networks and common applications. Topics include sentiment classification, information extraction, clustering, and topic modeling. Emphasis will be placed on the application of this technology to areas such as public health, crisis response, politics, and marketing.
Prerequisite(s): [(CS 430)]
(3-0-3)

CS 580
Topics in Machine Learning
This course covers advanced topics in machine learning. The exact course content may change based on recent advances in the area and the instructor teaching it. Possible topics include active learning, reinforcement learning, online learning, non-parametric learning, inductive learning, statistical relational learning, dimensionality reduction, ensemble methods, transfer learning, outlier detection, specific application areas of machine learning, and other relevant and/or emerging topics.
(3-0-3)

CS 581
Topics in Artificial Intelligence
Covers various advanced topics in AI, including both theory and practice. Content may vary by instructor. Possible topics include: Planning; STRIPS planning; Partial-order planning; Situation calculus; Theorem proving; GraphPlan/SatPlan; Transformational planning; Simulated annealing; Motion planning; Case-based reasoning; Multi-agent coordination; Negotiation planning; Representation and Reasoning; Logical representation; Frame problem; Probabilistic reasoning; Bayesian networks; Game Playing; Minimax search; Evaluation functions; Learning evaluation functions; Markov Decision Processes; Reinforcement learning for games; Developing AI agents; Multi-agent planning.
Prerequisite(s): [(CS 480)]
(3-0-3)

CS 582
Computational Robotics
Covers basic algorithms and techniques used in Computational Robotics, to give the student a good basis for work in this highly relevant field. Topics include: Locomotion, Non-visual sensors and algorithms, Uncertainty modeling, data fusion, State space models, Kalman filtering, Visual sensor, Sampling theory, Image features, Depth reconstruction, Multiple view geometry, Ego-motion, Active vision, Reasoning, Spatial decomposition, Geometric representations, Topological representations, Path planning, Spatial uncertainty, Active control, Pose maintenance, Dead reckoning, Correlation-based localization, Sensorial maps, Task planning and task interference, Multi-agent coordination.
Prerequisite(s): [(CS 430)]
(3-0-3)

CS 583
Probabilistic Graphical Models
This course will cover probabilistic graphical models – powerful and interpretable models for reasoning under uncertainty. The generic families of models such as directed, undirected, and factor graphs as well as specific representations such as hidden Markov models and conditional random fields will be discussed. The discussions will include both the theoretical aspects of representation, learning, and inference, and their applications in many interesting fields such as computer vision, natural language processing, computational biology, and medical diagnosis.
(3-0-3)

CS 584
Machine Learning
Introduce fundamental problems in machine learning. Provide understanding of techniques, mathematical concepts, and algorithms used in machine learning. Provide understanding of the limitations of various machine learning algorithms and the way to evaluate performance of learning algorithms. Topics include introduction, regression, kernel methods, generative learning, discriminative learning, neural networks, support vector machines, graphical models, unsupervised learning, and dimensionality reduction.
Prerequisite(s): [(CS 430)]
(3-0-3)

CS 585
Natural Language Processing
Prerequisite(s): [(CS 430)]
(3-0-3)

CS 586
Software Systems Architectures
This course covers the state-of-the-art in architectural design of complex software systems. The course considers commonly-used software system architectures, techniques for designing and implementing these architectures, models and notations for characterizing and reasoning about architectures, and case studies of actual software system architectures.
Prerequisite(s): [(CS 487)]
(3-0-3)

CS 587
Software Project Management
Concepts of software product and process quality. Role of TQM in software project management. Use of metrics, feasibility studies, cost and effort estimates. Discussion of project planning and scheduling. The project team and leadership issues. The Capability Maturity Model: basic tenets and application of process evaluation.
Prerequisite(s): [(CS 487)]
(3-0-3)

CS 588
Advanced Software Engineering Development
Software development process improvement is a major objective of this course. This is achieved through a series of individual programming and process projects. Students learn to plan their projects, apply measurements, estimate size, schedule tasks, and classify defects in order to improve the quality of both their development process and their software products.
Prerequisite(s): [(CS 487)]
(3-0-3)
CS 589
Software Testing & Analysis
Concepts and techniques for testing and analysis of software. Software testing at the unit, subsystem, and system levels. Specification-based testing. Code-based testing. Model-based testing. Methods for test generation and validation. Static and dynamic analysis. Formal methods and verification. Reliability analysis. Prerequisite(s): [(CS 487)] (3-0-3)

CS 590
Seminar in Computer Science
Investigation and discussion by faculty and students concentrated on some topic of current interest. May be taken more than once. Prerequisite: Instructor permission required. (3-0-3)

CS 591
Research & Thesis of Masters Degree
Instructor permission required. (Credit: Variable)

CS 594
Research Problems
Instructor permission required. (Credit: Variable)

CS 595
Topics in Computer Science
This course will treat a specific topic, varying from semester to semester, in which there is a particular student or staff interest. May be taken more than once. (Credit: Variable)

CS 597
Reading & Special Problems
May be taken more than once. (Credit: Variable) Instructor permission required. (Credit: Variable)

CS 612
Topics in Computer Vision
Covers advanced topics in computer vision to enhance knowledge of students interested in this highly important area. The topics in this course may change between semesters depending on the instructor teaching the course and the current state of the art in this area. Possible topics include: Image based modeling and rendering, Multiple view geometry, Auto-calibration, Object recognition, Motion analysis, Tracking, Perceptual user interfaces, Face and gesture recognition, Active vision. Prerequisite(s): [(CS 512)] (0-0-3)

CS 630
Advanced Topics in Algorithms
Theoretical analysis of various types of algorithms. Topics vary, and may include approximation, quantum, on-line, distributed, randomized, and parallel algorithms. Requires CS 430. Instructor permission required. Prerequisite(s): [(CS 430)] (3-0-3)

CS 642
Advanced Topics in Networking
Introduction to advanced networking research. A particular focus area will be considered, keeping current with advances in computer networking. Quantitative methods will be emphasized. Prerequisite(s): [(CS 542)] (3-0-3)

CS 681
Topics in Computational Linguistics
CS 585 Covers various topics in linguistics as they may be applied to various computational problems in AI, NLP, or IR. The topics in this course may change between semesters depending on the instructor teaching the course and the current state of the art in this area. Possible topics include: Systemic Functional Linguistics, Clausal structure, Group structure, Complex structure, Cognitive Linguistics, Process semantics. Prerequisite(s): [(CS 585)] (3-0-3)

CS 689
Advanced Topics in Software Engineering
Course content is variable and reflects the current trends in software engineering. Instructor permission required. (3-0-3)

CS 691
Research & Thesis Ph.D.
Instructor permission required. (Credit: Variable)

CS 695
Doctoral Seminar
Doctoral seminar. (1-0-1)

CS 750
Computer Aided Software Engineering
This course presents the state-of-the-art of computer-aided software engineering technologies. CASE encompasses a collection of automated tools and methods that provide automated support to the software specification, design, development, testing, maintenance, and management of large and complex software systems. Students will develop working understanding of CASE methodologies and tools. Prerequisite(s): [(CS 487)] (2-0-2)

CS 763
Automated Software Testing
This course will examine both the state-of-the-art and the state-of-practice in automated software testing on a system level and an unit level. Relevant issues include theoretical foundations of automated testing, automation tools and techniques, empirical studies and industrial experience. Key topics include, but are not limited to: Fundamentals of automated software testing, automated test design, modeling and generation, automated test execution, automated test management, automated test metrics, automated tools, automated feature and regression testing Environments to support cost-effective automated software testing, discussions on the barriers to industrial use of automated testing. Prerequisite(s): [(CS 487)] (2-0-2)
Computer Science Professional Master

CSP 527
Client-Server Applications Development
Through hands-on experience in developing a client-server database project and developing and managing a client-server Internet project, this course teaches advanced skills for effective design and implementation of client-server applications. Students will examine the architectural and functionality decisions, technologies, configurations, languages, and techniques associated with client-server systems. Active/passive client-server technologies, as well as public, enterprise-wide, and inter-enterprise approaches to decision and operation support are discussed and implemented.
Prerequisite(s): [(CS 425)]
(3-0-3)

CSP 541
Internet Technologies
This course focuses on the technologies and protocols used by Internet WAN’s and LAN’s. The fundamental architecture, organization, and routing principles of the Internet are described. Part of the course will focus on emerging Internet technologies.
Prerequisite(s): [(CS 455)]
(3-0-3)

CSP 542
Internet Design & Analysis
This course examines the principles for network design. The design process is studied from requirements gathering to deployment. The student will gain experience in estimating application load, network sizing, component choice, and protocol choice. Internetworking between popular components and protocols will be studied. Analytical and simulation techniques are described and used to design several local- and wide-area networks.
Prerequisite(s): [(CS 455)]
(3-0-3)

CSP 543
Multimedia Networking
This course covers the architectures, protocols, and design issues for multimedia networks. Topics covered include coding, compression, streaming, synchronization, QoS, and adaptation. Current tools for multimedia networking will be surveyed. Issues with multimedia application development will be explored. Students will design and develop multimedia applications.
Prerequisite(s): [(CS 455)]
(3-0-3)

CSP 544
System & Network Security
This course will present an in-depth examination of topics in data and network security such as: Access control, authentication, security assessment, network and data security tools, and security policies. A significant hands-on component includes network incidents to detect and fix.
Prerequisite(s): [(CS 430 and CS 455)]
(3-0-3)

CSP 545
Wireless Networking Technologies & Applications
This course will present the foundation of wireless technologies and examine state-of-the-art wireless systems, services, network technologies, and security.
Prerequisite(s): [(CS 542)]
(3-0-3)

CSP 550
Internet Programming
This course discusses current fundamental concepts and development techniques for distributed applications. Topics covered include multithreaded programs, sockets, message-passing systems, remote method invocation and procedure calls, peer-to-peer networks, and underlying technologies for internet applications.
Prerequisite(s): [(CS 450)]
(3-0-3)

CSP 551
Advanced UNIX Programming
This course provides a hand-on introduction to UNIX programming topics such as standard application programmer interfaces, concurrent programming, UNIX processes and threads, shell programming, UNIX interprocess communications, client-server designs, and application portability.
Prerequisite(s): [(CS 450)]
(3-0-3)

CSP 570
Data Science Seminar
This required seminar course surveys current applications of data science, bringing in lecturers from industry and academia to discuss real-world problems and how they are addressed within a data analytic framework. Students are required to attend all lectures and to give a short presentation or paper on one of the topics at the end of the semester. Permission is required from the instructor or department. Open only to Data Science majors.
(1-0-0)

CSP 571
Data Preparation & Analysis
Surveys industrial and scientific applications of data analytics with case studies including exploration of ethical issues via case studies. Students will work with a variety of real world data sets and learn how to prepare data sets for analysis by cleaning and reformatting. We will also cover a variety of data exploration techniques including summary statistics and visualization methods. Open only to Data Science majors.
(3-0-3)

CSP 572
Data Science Practicum
Students will work in small groups to solve real-world data analysis problems for actual scientific or industrial clients. Innovation and clarity of presentation will be key elements of evaluation. Students will also have an option to fulfill course requirements through a data analytics internship with an industry partner. Open only to Data Science majors.
(0-0-6)

CSP 581
Applied Artificial Intelligence Programming
To learn AI programming algorithms and techniques in common lisp. Time is split between common Lisp topics and discussions of implementation strategies for AI algorithms.
Prerequisite(s): [(CS 440)]
(3-0-3)
CSP 585  
**Object-Oriented Design Patterns**  
This course introduces the principles of design patterns for Object-Oriented software systems. A catalog of design patterns is shown, to illustrate the roles of patterns in designing and contracting complex software systems. The catalog of design patterns also provides a pragmatic reference to a well-engineered set of existing patterns currently in use. Also discussed is the impact of post-object oriented software development on design patterns.  
Prerequisite(s): [(CS 445)]  
(3-0-3)  

CSP 586  
**Software Modeling Development with UML**  
Students will obtain a significant exposure to the UML technology. This will include exposure to modeling, model-driven development, executable models, and round-trip engineering.  
Prerequisite(s): [(CS 445) OR (CS 487)]  
(3-0-3)  

CSP 587  
**Software Quality Management**  
Students will learn methods of software quality management. This will include exposure to software quality assurance, quality measures, and quality control. These quality management methods will be explained at the applications level.  
Prerequisite(s): [(CS 487)]  
(3-0-3)  

CSP 595  
**Topics in Computer Science Professional Master**  
Topics in CS.  
(3-0-3)  

Undergraduate Courses Available to Graduate Students  
Note: Students may take up to an approved number of the following courses.  
**CS 411**  
Computer Graphics  
**CS 422**  
Data Mining  
**CS 425**  
Database Organization  
**CS 429**  
Information Retrieval  
**CS 430**  
Information Retrieval  
**CS 440**  
Programming Languages and Translators  
**CS 441**  
Programming Languages and Translators  
**CS 445**  
Object Oriented Design and Programming  
**CS 447**  
Distributed Objects  
**CS 450**  
Operating Systems  
**CS 455**  
Data Communications  
**CS 458**  
Information Security  
**CS 470**  
Computer Architecture  
**CS 480**  
Artificial Intelligence Planning and Control  
**CS 485**  
Computers and Society  
**CS 487**  
Software Engineering I