Computer and Information Science

Joseph Sventek, Department Head
541-346-4408
541-346-5373 fax
120C Deschutes Hall
1202 University of Oregon
Eugene OR 97403-1202

Computer science, the study of computation, offers students the challenge and excitement of a dynamically evolving science whose discoveries and applications affect every aspect of modern life. Computer science is a rich intellectual field where practitioners apply a computational approach to address a wide variety of interesting and challenging problems. Computer scientists are engaged in research in core areas of theoretical computer science, computer systems design, algorithms, and programming languages, as well as more application-oriented areas such as databases, networking, and informatics.

The Department of Computer and Information Science (CIS) is committed to a strong research program and a rewarding educational experience for undergraduate and graduate students.

The department offers instruction and opportunities for research in the following areas:

- theoretical computer science
- computational science
- scientific visualization and high-performance computing
- human-computer interaction, visualization
- computer security
- software engineering
- networking
- parallel and distributed computing
- programming languages and compilers
- artificial intelligence
- informatics
- universal access

The department offers bachelor's, master's and doctoral degrees; in addition, two undergraduate minors and a selection of service courses are offered for students who want introductory exposure to computers and computer applications. The computer science programs at the university are continually evolving as the discipline matures and as students' needs change.

Facilities

The Department of Computer and Information Science is housed in Deschutes Hall, which holds faculty and graduate student offices and extensive laboratory space for research and instruction.

Undergraduate majors may use campus computing labs staffed by CIS undergraduate tutors and lab assistants. Undergraduate majors taking upper-division courses and graduate students share a collaborative computing lab for the exclusive use of CIS students. Graduate and undergraduate students engaged in active research also have access to the computing facilities of the associated research lab.

The cognitive modeling and eye-tracking laboratory features multiple Eyegaze eye trackers and a Tobii eye tracker, used to collect and analyze the eye movements people make during human-computer interactions, and to develop eye-controlled user interfaces for people with disabilities. The lab also features a VRSonic SoundSim Cube spatialized audio server used to explore three-dimensional sound perception.

Research in high-performance computing and computational science is supported by resources shared among the department, the Computational Science Institute, and the Neuroinformatics Center. In addition, the University of Oregon, through a Major Research Instrumentation Program grant from the National Science Foundation, has created a large-scale computational and storage resource to support multidisciplinary scientific research. The Applied Computational Instrument for Scientific Synthesis (ACISS) is a heterogenous platform managed as a cloud system for computational science, informatics, and data science.

The Advanced Integration and Mining Laboratory fosters research on finding useful patterns from the mountain of data on biology, health, medicine, neuroscience, physiology, and social networks and on integrating data from structurally and semantically heterogeneous resources such as databases, online social networks, and the World Wide Web.

The Oregon Network Research Group Laboratory features an array of high-end servers, experimental test beds, and mass storages for developing and testing of new network protocols, conducting large-scale network measurement and data analysis.

The Network Security Laboratory features hardware and software facilities devoted to experimentation, simulation, and analysis of various computer networking techniques (such as Internet routing, software-defined networking, online social networking, and cloud networking), malicious network attacks (such as distributed denial-of-service attacks, traffic route hijacking, Internet worms, botnets, Sybil attacks), and cyberdefense technologies (such as firewalls, antiphishing solutions, distributed denial-of-service defense, IP spoofing prevention, Internet routing security, Internet privacy protection, and Internet of Things security and privacy).

The Ubiquitous Computing Laboratory uses a mix of custom-designed and commercial hardware to study the application of assistive software to everyday living. Researchers are particularly interested in the design of software for those with impairments that limit their use of commercial, off-the-shelf software.

The Research Group on Computing and Data-Understanding at Extreme Scale pursues problems in scientific visualization, high-performance computing, scientific computing, information visualization, and computer graphics, and especially focuses on problems where these areas intersect. The group performs research for the Department of Energy, the National Science Foundation, and private companies, delivered in widely used software tools such as the VisIt visualization tool, and helps develop new tools, like GraviT, a distributed-memory parallel ray-tracing library.

The High-Performance Computing Laboratory conducts research in several areas, including static analysis of software for building performance models and detecting security vulnerabilities, source-to-source approaches for semantics-preserving (e.g., performance optimization) and semantics-modifying (e.g., security-vulnerability fixes, automatic differentiation) transformations. The lab also performs research in modeling run-time characteristics of software, and developing and

employing numerical optimization techniques for maximizing multiple run-time objectives (performance, energy efficiency, resilience).

Software engineering is applied to two emerging areas: data science and the Internet of Things. The Flare Project is exploring new tools for data scientists. The project involves working with data science frameworks (e.g. Galaxy), new workflow programming languages, and open cloud software (e.g., OpenStack) to support it all. The Foundry Project is exploring software frameworks to support the Internet of Things with a specific emphasis on reworking distributed algorithms that were devised in an era when cyber-security was less of a threat.

In addition, the university is a member of Internet2, a high-speed network connecting major research institutions.

Careers

The undergraduate program is designed to prepare students for professional careers or graduate study. The field of computer science, which has become increasingly interdisciplinary over the past decade, offers a rich array of opportunities in fields as disparate as medicine, manufacturing, and the media as well as the computer industry.

Graduates come away with confidence that they can specify, design, and build large software systems; analyze the effectiveness of computing techniques for a specific problem; and work effectively in problem-solving teams. The master of arts (MA) and master of science (MS) degree programs prepare students for higher-level positions in the areas described above as well as for teaching positions in community colleges. The PhD degree program trains students as scientists for advanced research in specialized areas of computer science and for teaching in universities.

Faculty

Zena M. Ariola, professor (programming languages, lambda calculus, logic). BS, 1980, Pisa; PhD, 1992, Harvard. (1992)

Hank Childs, associate professor (scientific visualization, high-performance computing, computer graphics). BS, 1999, PhD, 2006, California, Davis. (2013)

Dejing Dou, professor (artificial intelligence, information integration, data mining). BE, 1996, Tsinghua; MS, 2000, PhD, 2004, Yale. (2004)

Stuart Faulk, instructor. BA 1973, Cornell; PhD, 1989, North Carolina, Chapel Hill. (1989)

Stephen F. Fickas, professor (requirements engineering, software engineering, wearable ubiquitous computing). BS, 1971, Oregon State; MS, 1973, Massachusetts; PhD, 1982, California, Irvine. (1982)

Kathleen Freeman Hennessy, senior instructor; director of undergraduate studies. BS, 1982, Bucknell; PhD, 1993, Oregon. (2011)

Michael Hennessy, senior instructor. BS, 1982, MS, 1984, Oregon. (1984)

Anthony J. Hornof, associate professor (human-computer interaction, cognitive modeling, eye tracking). BA, 1988, Columbia; MS, 1996, PhD, 1999, Michigan. (1999)

Jun Li, professor (computer and network security, network architectures and protocols, distributed systems). BS, 1992, Peking; ME, 1995, Chinese Academy of Sciences; MS, 1998, PhD, 2002, California, Los Angeles. (2002)

Daniel Lowd, assistant professor (machine learning, data mining, artificial intelligence). BS, 2003, Harvey Mudd College; MS, 2005, PhD, 2010, Washington (Seattle). (2009)

Allen D. Malony, professor (parallel processing, performance evaluation, neuroinformatics). BS, 1980, MS, 1982, California, Los Angeles; PhD, 1990, Illinois, Urbana-Champaign. (1991)

Boyana Norris, associate professor (high-performance computing, source-code analysis and transformation) BS, 1995, Wake Forest; PhD, 2000, Illinois, Urbana-Champaign. (2013)

Reza Rejaie, professor (computer networks, multimedia networking, network measurement). BS, 1991, Sharif University of Technology; MS, 1996, PhD 1999, Southern California. (2002)

Joseph Sventek, professor (complex event processing, Internet of Things). BS, 1973, Rochester; PhD, 1979, California, Berkeley. (2014)

Dave Wilkins, instructor. BA, 1965, Whitman College; MS, 1971, Oregon (2010)

Eric D. Wills, senior instructor. BS, 2000, MS, 2002, PhD, 2008, Oregon. (2010)

Christopher B. Wilson, associate professor (computational complexity, models of computation). BS, 1978, Oregon; MS, 1980, PhD, 1984, Toronto. (1984)

Xiaodi Wu, assistant professor (quantum information and computation, cryptography, theory of computation). BS, 2008, Tsinghua; PhD, 2013, Michigan, Ann Arbor. (2015)

Michal Young, associate professor (software engineering, software test and analysis). BA, 1983, Oregon; MS, 1985, PhD, 1989, California, Irvine. (1997)

Emeriti

Sarah A. Douglas, professor emerita. AB, 1966, California, Berkeley; MS, 1979, PhD, 1983, Stanford. (1983)

Arthur M. Farley, professor emeritus. BS, 1968, Rensselaer Polytechnic Institute; PhD, 1974, Carnegie-Mellon. (1974)

Virginia M. Lo, associate professor emerita. BA, 1969, Michigan; MS, 1977, Pennsylvania State; PhD, 1983, Illinois, Urbana-Champaign. (1984)

Eugene M. Luks, professor emeritus. BS, 1960, City University of New York, City College; PhD, 1966, Massachusetts Institute of Technology. (1983)

Andrzej Proskurowski, professor emeritus. MS, 1967, Warsaw Technical; PhD, 1974, Royal Institute of Technology, Stockholm. (1975)

Kent A. Stevens, professor emeritus. BS, 1969, MS, 1971, California, Los Angeles; PhD, 1979, Massachusetts Institute of Technology. (1982)

The date in parentheses at the end of each entry is the first year on the University of Oregon faculty.

Participating

John S. Conery, biology

Amy K. Lobben, geography

Jeffrey Stolet, music

Joseph W. Thornton, biology

Don M. Tucker, psychology

- **Bachelor of Arts**
- **Bachelor of Science**
- **Computer and Information Science Minor**
- **Computer Information Technology Minor**

Undergraduate Studies

The Department of Computer and Information Science offers a major and a minor in computer and information science, a major in mathematics and computer science, and a minor in computer information technology.

The computer and information science major is intended for students who want to study computers and computation with strong mathematical and scientific foundations. The mathematics and computer science major emphasizes formal and abstract problem solving complemented by computational methods and computer technologies. This program, administered jointly with the Department of Mathematics, is described in the **Mathematics and Computer Science** section of this catalog. Both of these majors lead to the bachelor of arts (BA) or bachelor of science (BS) degrees.

Students majoring in computer and information science may choose to focus their studies in one of several areas of specialization, or tracks, which build on the standard CIS core requirements. Each track specifies a set of coordinated choices for fulfilling upper-division computer science and other elective requirements. Students interested in specializing in a track should contact their advisor as early as possible. Current computer science tracks include foundations, software development, computer networks, database and informatics, computational arts: multimedia, business information systems, computational science, and security.

Preparation

High school students who plan to major in computer and information science should pursue a strong academic program, including substantial work in mathematics and the sciences. Courses in algebra, geometry, trigonometry, and more advanced topics should be included. Substantial experience in expository and technical writing is highly desirable. Courses in computer programming or computer technology are useful but not required. Upon arrival at the university, freshmen should consult with a CIS advisor to find the entry-level course best suited to the student's background.

Transfer and Second Baccalaureate Students

Transfer and second baccalaureate students should consult the online Interactive Transfer Catalog as well as a CIS advisor to determine whether computer science, mathematics, and science courses they have taken fulfill the major requirements. Completing only general-university requirements prior to transferring to the University of Oregon will not be sufficient preparation to complete a CIS degree in two years.

Students attending community college in Oregon are encouraged to obtain the associate of arts Oregon transfer degree or the associate of science Oregon transfer degree in computer science before entering the University of Oregon. While earning this degree, community college transfer students should take discrete mathematics and computer science. In addition, calculus and laboratory science are recommended.

Bachelor of Arts Degree Requirements

To earn a BA in computer and information science, majors must complete the requirements for a BS in addition to demonstrating proficiency in a second language. Computer and information science majors must complete at least 60 credits of CIS courses, of which 24 must be earned in residence at the University of Oregon. In addition, majors must complete 28 credits in mathematics, 12 credits in the sciences, and 4 credits of technical or business writing. The specific requirements for the CIS major fall into five categories: core courses, track and elective courses, mathematics, writing, and science. Core courses in the lower division must be passed with a grade of B– or better for automatic advancement to the upper-division core courses. Upper-division core courses must be passed with a grade of C– or better.

Code	Title	Credits
Core Courses: Lower Division		
CIS 210–212	Computer Science I-III	12
MATH 231–232	Elements of Discrete Mathematics I-II	8
Core Courses: Upper Division		
CIS 313	Intermediate Data Structures	4
CIS 314	Computer Organization	4
CIS 315	Intermediate Algorithms	4
CIS 330	C/C++ and Unix	4
CIS 415	Operating Systems	4
CIS 422	Software Methodology I	4
CIS 425	Principles of Programming Languages	4
Core Courses: Mathematics		
Select one of the following:		8
MATH 251–252	Calculus I-II	
MATH 261–262	Calculus with Theory I-II	
MATH 246–247	Calculus for the Biological Sciences I-II	
Select two of the following:		8
MATH 233	Elements of Discrete Mathematics III	
MATH 253	Calculus III	
or MATH 263	Calculus with Theory III	
MATH 341	Elementary Linear Algebra	
MATH 343	Statistical Models and Methods	
or MATH 425	Statistical Methods I	
Core Courses: Science		
Select 12 credits from the following: [1]		12
Biology [2]		
CH 111	Introduction to Chemical Principles	
or CH 113	The Chemistry of Sustainability	
or CH 221	General Chemistry I	
or CH 224H	Honors General Chemistry	
BI 211,213	General Biology I,III	

or BI 211–212	General Biology I-II	
Chemistry [2]		
CH 221–223	General Chemistry	
or CH 224H–226H	Honors General Chemistry	
Earth Sciences		
GEOL 201	Earth's Interior Heat and Dynamics	
GEOL 202	Earth Surface and Environmental Geology	
GEOL 203	Evolution of the Earth	
Geography		
GEOG 141	The Natural Environment	
Select two of the following:		
GEOG 321	Climatology	
GEOG 322	Geomorphology	
GEOG 323	Biogeography	
Physics [2,3]		
PHYS 201–203	General Physics	
or PHYS 251–253	Foundations of Physics I	
Psychology		
PSY 201	Mind and Brain	
PSY 202	Mind and Society	
Select one of the following:		
PSY 304	Biopsychology	
PSY 348	Music and the Brain	
Core Course: Writing		
WR 320	Scientific and Technical Writing	4
or WR 321	Business Communications	
Electives: Upper Division		
Upper-division CIS courses in student's chosen track (track information below)		12
Upper-division CIS courses in student's chosen track, honors thesis, capstone project, or other upper-division courses [4]		8
Upper-division mathematics or theoretical computer science course [5]		4
Total Credits		**104**

1 To support interdisciplinary study, students on any track are encouraged to complete a minor (typically 24–32 credits) or major in another field . Students who complete a minor or major in another field (other than in CIT or mathematics) may, with advisor approval, replace the CIS lab science requirement with the completed minor or major.

2 Students are encouraged to complete the accompanying lab courses.

3 Physics is recommended for networks track students.

4 A maximum of 8 credits in courses numbered 399-409. None of these courses can be taken for more then 4 credits when used to satisfy this requirement.

5 The mathematics elective is selected from mathematics courses with a prerequisite of Calculus II (MATH 252) or higher, or from theoretical computer science courses. A list of theoretical computer science courses is available in the computer science office or the department website.

Bachelor of Science Degree Requirements

To earn a BS in computer and information science, majors must complete at least 60 credits of CIS courses, of which 24 must be earned in residence at the University of Oregon. In addition, majors must complete 28 credits in mathematics, 12 credits in the sciences, and 4 credits of technical or business writing. The specific requirements for the CIS major fall into six categories: core courses, track courses, electives, mathematics, writing, and science. Core courses in the lower division must be passed with a grade of B– or better for automatic advancement to the upper-division core courses. Upper-division core courses must be passed with a grade of C– or better.

Code	Title	Credits
Core Courses: Lower Division		
CIS 210–212	Computer Science I-III	12
MATH 231–232	Elements of Discrete Mathematics I-II	8
Core Courses: Upper Division		
CIS 313	Intermediate Data Structures	4
CIS 314	Computer Organization	4
CIS 315	Intermediate Algorithms	4
CIS 330	C/C++ and Unix	4
CIS 415	Operating Systems	4
CIS 422	Software Methodology I	4
CIS 425	Principles of Programming Languages	4
Core Courses: Mathematics		
Select one of the following:		8
MATH 251–252	Calculus I-II	
MATH 261–262	Calculus with Theory I-II	
MATH 246–247	Calculus for the Biological Sciences I-II	
Select two of the following:		8
MATH 233	Elements of Discrete Mathematics III	
MATH 253	Calculus III	
or MATH 263	Calculus with Theory III	
MATH 341	Elementary Linear Algebra	
MATH 343	Statistical Models and Methods	
or MATH 425	Statistical Methods I	
Core Courses: Science		
Select 12 credits from the following: [1]		12
Biology [2]		
CH 111	Introduction to Chemical Principles	
or CH 113	The Chemistry of Sustainability	
or CH 221	General Chemistry I	
or CH 224H	Honors General Chemistry	
BI 211,213	General Biology I,III	
or BI 211–212	General Biology I-II	
Chemistry [2]		
CH 221–223	General Chemistry	
or CH 224H–226H	Honors General Chemistry	
Earth Sciences		
GEOL 201	Earth's Interior Heat and Dynamics	

GEOL 202	Earth Surface and Environmental Geology	
GEOL 203	Evolution of the Earth	
Geography		
GEOG 141	The Natural Environment	
Select two of the following:		
GEOG 321	Climatology	
GEOG 322	Geomorphology	
GEOG 323	Biogeography	
Physics [2,3]		
PHYS 201–203	General Physics	
or PHYS 251–253	Foundations of Physics I	
Psychology		
PSY 201	Mind and Brain	
PSY 202	Mind and Society	
Select one of the following:		
PSY 304	Biopsychology	
PSY 348	Music and the Brain	
Core Course: Writing		
WR 320	Scientific and Technical Writing	4
or WR 321	Business Communications	
Electives: Upper Division		
Upper-division CIS courses in student's chosen track (track information below)		12
Upper-division CIS courses in student's chosen track, honors thesis, capstone project, or other upper-division courses [4]		8
Upper-division mathematics or theoretical computer science course [5]		4
Total Credits		**104**

1. To support interdisciplinary study, students on any track are encouraged to complete a minor (typically 24–32 credits) or major in another field . Students who complete a minor or major in another field (other than in CIT or mathematics) may, with advisor approval, replace the CIS lab science requirement with the completed minor or major.
2. Students are encouraged to complete the accompanying lab courses.
3. Physics is recommended for networks track students.
4. A maximum of 8 credits in courses numbered 399-409. None of these courses can be taken for more then 4 credits when used to satisfy this requirement.
5. The mathematics elective is selected from mathematics courses with a prerequisite of Calculus II (MATH 252) or higher, or from theoretical computer science courses. A list of theoretical computer science courses is available in the computer science office or the department website.

Upper-Division Electives

In addition to the core CIS, mathematics, science, and writing courses, computer and information science majors must complete 20 credits of upper-division computer science and 4 credits of upper-division mathematics or theoretical computer science. Students complete at least 12 of their upper-division CIS credits with courses from their selected track; the 8 remaining upper-division credits may be additional track courses, an honors thesis, capstone project, or upper-division electives.

A maximum of 8 credits in CIS courses, numbered 399–409, may be applied to the upper-division electives requirement. Courses numbered 400–409 may be taken for a maximum of 4 credits when used to satisfy this requirement. Special Studies: [Topic] (CIS 399) and Experimental Course: [Topic] (CIS 410) courses used as upper-division electives must have a prerequisite of CIS 313 and have regular weekly class meetings and homework assignments.

The mathematics elective is selected from mathematics courses with a prerequisite of MATH 252 or higher, or from theoretical computer science courses. A list of courses is available in the computer science office or the department website.

Tracks

Tracks highlight areas of specialization within the department and guide student elective choices. Each track has an approved list of CIS courses, available from the computer science office or the department website. Tracks may also include recommended science or mathematics courses or a recommended minor in another field.

Foundations Track

The foundations track is the most general track, allowing a student to choose a set of electives tailored to his or her interests and intended choice of career.

Software Development Track

The software development track prepares students for careers in software engineering, software project management, software quality assurance, and other areas involving the creation of software. Course work focuses on solving problems related to the cost of development as well as the quality of the software delivered in complex software projects.

Computer Networks Track

The computer networks track prepares students for careers as network systems administrators, network protocol developer-programmers, or network security specialists in a wide range of environments, including educational institutions, business enterprises, and government agencies, as well as for advanced graduate studies and research in the field of computer networks. Course work encompasses most aspects of network theory and practice.

Database and Informatics Track

The database and informatics track prepares students for careers in database application programming, database design, doctoral work in business administration, and graduate work in informatics and database theory. Course work includes data structures, data architecture, and data mining.

Computational Arts: Multimedia Track

The computational arts: multimedia track, designed for computer and information science majors who plan to work in the field of multimedia arts, prepares the student for artistic and technical aspects of this area.

Business Information Systems Track

Graduates in the business information systems track are qualified to work as analysts, managers, developers, or consultants, and to enter leadership-development programs. Completion of this track, combined with professional work experience and economics courses, prepares

students to enter the Lundquist College of Business MBA program at the University of Oregon, and MBA programs at other universities.

Computational Science Track

The computational science track prepares students to apply computational and mathematical techniques to the analysis and management of biological data. Course work in this track combines depth in applied and formal aspects of computer science with rigorous training in biology.

Security Track

The security track provides a foundation in topics and concepts relating to the security of computer systems and networks. It prepares students to work as security analysts and provides a highly desirable skill set for all employers, ranging from software engineers to administrators, in both the private and government sectors. It also provides a foundation for further graduate study and research in security. Course work encompasses a strong understanding of computer systems and networks and their security, and can be tailored to a more theoretical or more applied focus.

Preparation for the Major

Students who take Computer Science I (CIS 210) are expected to have completed Elementary Functions (MATH 112) or the equivalent and to have prior programming experience from a high school course, through employment, or in a course such as Introduction to Programming and Problem Solving (CIS 122). Students who are unsure about their level of programming preparation should meet with a CIS advisor.

Sequence of Courses for Students with Programming Experience and Mathematical Background

Course	Title	Credits
First Year		
MATH 231–232	Elements of Discrete Mathematics I-II	8
CIS 210–212	Computer Science I-III	12
	Credits	**20**
	Total Credits	**20**

Sequence of Courses for Students without Programming Experience and Mathematical Background

Course	Title	Credits
First Year		
Fall		
MATH 112	Elementary Functions	4
Select one of the following: [1]		4
CIS 105	Explorations in Computing	4
CIS 122	Introduction to Programming and Problem Solving	4
CIS 111	Introduction to Web Programming	4
	Credits	**8**
Winter		
MATH 231–232	Elements of Discrete Mathematics I-II [2]	8
CIS 210–212	Computer Science I-III [2]	12
	Credits	**20**
	Total Credits	**28**

1 Taken either in first term or first year. Students are encouraged to take more then one course. CIS 122 strongly recommended.

2 Taken in either second term or second year.

Major Progress Review and Major in Good Standing

Each major must meet with a computer science advisor and file the Major Progress Review form while taking Intermediate Data Structures (CIS 313) or Computer Organization (CIS 314). Mathematics and CIS core courses and at least 12 credits of CIS upper division elective courses numbered 410 or higher must be taken for letter grades and passed with grades of C– or better. Other courses used to satisfy the major requirements may be taken for letter grades or pass/no pass. Grades of C– or better or P must be earned in these courses. At least 12 of the CIS upper-division credits applied to the degree must be taken in residence at the university. A student who receives two grades below C– in the upper-division core, or three grades below C– in any CIS upper-division courses, may be removed from the major.

Mathematics and Computer Science

The Department of Computer and Information Science and the Department of Mathematics jointly offer an undergraduate major in mathematics and computer science, leading to a bachelor of arts or a bachelor of science degree. This program is described in the Mathematics and Computer Science section of this catalog. This major prepares students for a wide range of careers in the high tech industry, for advanced graduate study, and for careers as middle school and high school teachers.

Honors Program

Students with a GPA of 3.50 or higher in computer and information science and a cumulative GPA of 3.50 or higher are encouraged to apply to the department honors program after completing Intermediate Data Structures (CIS 313), Computer Organization (CIS 314), Intermediate Algorithms (CIS 315), and C/C++ and Unix (CIS 330). The application form is available in the department office. To graduate with departmental honors, a student must write a thesis under the supervision of a faculty member.

Internships

Practical work experience in the software industry is seen as a valuable complement to academic course work. The department works with students to place them in internship positions in the summer and throughout the academic year. Students may also use the services of the UO Career Center and other agencies to identify internship opportunities. Majors may receive academic credit for internships. To earn upper-division elective credit for an internship, the work experience must be at a technical level beyond Intermediate Data Structures (CIS 313) and be sponsored by a CIS faculty member. A contract signed by the faculty sponsor, internship supervisor, and the student must be filed with the department before the internship begins.

Research

Faculty members in the computer and information science department receive grants from government, industry, and private sources to

conduct research in their areas of expertise. Undergraduate majors are encouraged to take part in the various research groups in the department. Most students begin approaching faculty members for such opportunities while taking the 300-level courses. Research can be used to fulfill upper-division electives, as part of an honors thesis, or in some cases as a paid internship.

Awards and Honor Societies

The Erwin and Gertrude Juilfs Scholarship in Computer and Information Science, in honor of Erwin and Gertrude Juilfs, is awarded to one or more students who show exceptional promise for achievement as evidenced by grade point average, originality of research, or other creative activities.

The Geoffery Eric Wright Outstanding Junior Award, in honor of CIS student Geoffery Wright, is a scholarship for students displaying high-quality academic performance, commitment to learning, and a promise of further outstanding achievement in computer and information science and its applications.

The J. Donald Hubbard Scholarship in Computer and Information Science, in honor of J. Donald Hubbard, recognizes an undergraduate or graduate student who shows outstanding promise in the fields of computer-human interaction, computer graphics, or multimedia.

The Phillip Seeley Scholarship in Computer and Information Science has been established as a permanent endowment to provide a source of income supporting a scholarship for outstanding undergraduate CIS students. This scholarship is based on overall quality of academic work, commitment to learning, and potential for further academic achievement. Preference is given to resident Oregon students with financial need, as determined by the UO Office of Student Financial Aid and Scholarships.

Students with outstanding academic accomplishments may be invited to become members of Upsilon Pi Epsilon, the international honor society in computer science.

Minor Requirements

Computer and Information Science Minor

The minor in computer and information science introduces the theories and techniques of computer science and develops programming skills that are applicable to the student's major. It is a strong complement to a major in any of the sciences and in related fields such as multimedia arts. Students from all majors have found their career opportunities enhanced through the CIS minor.

Before enrolling in upper-division courses, students planning a minor in computer and information science must file an application form with the department. Each student should consult with a CIS faculty advisor to plan the minor program.

Courses applied to the CIS minor must be completed with grades of C– or better.

Lower-Division Courses		
CIS 210–212	Computer Science I-III	12
MATH 231–232	Elements of Discrete Mathematics I-II	8
Upper-Division Courses		
CIS 313	Intermediate Data Structures	4
Electives [1]		8
Total Credits		**32**

[1] Classes number CIS 399-409 with departmental permission only

Computer Information Technology Minor

The minor in computer information technology (CIT) prepares students to work with evolving technologies for work environments that require development and management of web applications, databases, computer networks, open-source platforms, and cloud computing. It provides practical experience in understanding the tools and technologies of the computing field. It goes well with majors in the professional schools such as business and journalism and is an excellent match with almost any major on campus.

Before enrolling in CIT upper-division courses, students planning a minor in computer information technology must file an application form with the department. Each student should consult with an assigned CIT faculty advisor to plan the minor program.

Lower-division courses must be completed with grades of B– or better. Upper-division courses must be taken in sequence and are offered only once a year. Upper-division courses must be completed with grades of C– or better.

Lower-Division Courses		
Option 1:		12
CIS 110	Fluency with Information Technology	
CIS 111	Introduction to Web Programming	
CIT 281	Web Applications Development I	
Option 2:		
CIS 111	Introduction to Web Programming	
CIS 115	Multimedia Web Programming	
CIT 281	Web Applications Development I	
Upper-Division Courses		
CIT 381	Database Systems	4
CIT 382	Web Applications Development II	4
CIT 383	Networking Fundamentals	4
Total Credits		**24**

- **Master of Arts**
- **Master of Science**
- **Doctor of Philosophy**

Graduate Studies

The department offers programs leading to the master of arts (MA), master of science (MS), and doctor of philosophy (PhD).

Master's Degree Program

Admission

Admission to the master's degree program in computer and information science is competitive. It is based on prior academic performance, Graduate Record Examinations (GRE) scores, and computer science background. Minimum requirements for admission with graduate master's status are as follows:

1. Documented knowledge of
   a. Principles of computer organization and operating systems
   b. Programming languages
   c. Program development and analysis

   d. Data structures and algorithm analysis
2. GRE scores on the general test. The computer science test is optional.
3. A score of at least 100 on the Internet-based option of the Test of English as a Foreign Language (TOEFL iBT) or a score of 7.0 on the International English Language Testing System (IELTS) for applicants who have no justification for a waiver. Applicants may be required to study one or more terms at the university's American English Institute or elsewhere before taking any graduate work in the department. International applicants for teaching assistantships who score at least 26 on the speaking section of the TOEFL iBT will not have to take the Speaking Proficiency English Assessment Kit (SPEAK) test upon arrival at the university.
4. Three letters of recommendation, a statement of purpose, and unofficial transcripts (via on online application). Note that official transcripts are sent to the UO Office of Admissions.

Grades from previous course work should indicate the ability to maintain at least a 3.00 grade point average in graduate-level courses.

Application materials should be submitted by December 15 for admission, via GradWeb (http://gradweb.uoregon.edu), the following fall term.

Admission to the master's degree program requires the substantive equivalent of an undergraduate degree in computer science. A second bachelor's degree program can be used to gain the required level of computer science background. Students without this background may be admitted conditionally and required to complete remedial course work before achieving unconditional standing in the program.

Master of Arts Degree Requirements

Code	Title	Credits
Core Courses		
CIS 621	Algorithms and Complexity	4
CIS 624	Structure of Programming Languages	4
CIS 630	Distributed Systems	4
Cluster Courses		
Required course [1]		4
Two depth courses from cluster [1]		8
Electives		
Select 30 credits from the following:		30
CIS 601 & CIS 605	Research: [Topic] and Reading and Conference: [Topic] [2]	
CIS 503	Thesis (plus 4 credits from CIS 601–608 for either course selected) [3]	
or CIS 609	Final Project	
CIS 510	Experimental Course: [Topic] [4]	
or CIS 610	Experimental Course: [Topic]	
CIS 601–CIS 608 [5]		
Twelve credits in courses outside department in area closely related to professional goals [6]		
Departmental courses numbered 500 or higher		
Total Credits		**54**

1. A list of clusters is available in the department office.
2. Up to 8 credits.
3. Up to 12 credits.
4. Must receive approval from the graduate education committee.
5. Up to 8 credits for master's students who do not complete a thesis or final project.
6. Courses must be approved by the student's academic advisor; options include courses in linguistics, mathematics, physics, and psychology.

Master of Science Degree Requirements

Code	Title	Credits
Core Courses		
CIS 621	Algorithms and Complexity	4
CIS 624	Structure of Programming Languages	4
CIS 630	Distributed Systems	4
Cluster Courses		
Required course [1]		4
Two depth courses from cluster [1]		8
Electives		
Select 30 credits from the following:		30
CIS 601 & CIS 605	Research: [Topic] and Reading and Conference: [Topic] [2]	
CIS 503	Thesis (plus 4 credits from CIS 601–608 for either course selected) [3]	
or CIS 609	Final Project	
CIS 510	Experimental Course: [Topic] [4]	
or CIS 610	Experimental Course: [Topic]	
CIS 601–CIS 608 [5]		
Twelve credits in courses outside department in area closely related to professional goals [6]		
Departmental courses numbered 500 or higher		
Total Credits		**54**

1. A list of clusters is available in the department office.
2. Up to 8 credits.
3. Up to 12 credits.
4. Must receive approval from the graduate education committee.
5. Up to 8 credits for master's students who do not complete a thesis or final project.
6. Courses must be approved by the student's academic advisor; options include courses in linguistics, mathematics, physics, and psychology.

Complex Software Systems

Students must show competency in the design and implementation of software systems by taking one course that requires a substantial programming project. A list of courses that satisfy this requirement is available in the department office.

Grade Requirements

The 24 credits in the required courses and the cluster courses must be passed with grades of B– or better. As many as 12 of the 30 elective credits may be taken pass/no pass (P/N); graded elective courses must be passed with grades of C or better. A 3.00 GPA must be maintained for courses taken in the program.

Master's Thesis

The research option requires a written thesis and 9 to 12 credits in Thesis (CIS 503). Thesis research is supervised by a faculty advisor; this advisor

and other faculty members constitute the thesis committee. The master's thesis is expected to be scholarly and to demonstrate mastery of the practices of computer science. This option is recommended for students who plan subsequent PhD research.

Master's Project

The project option requires a minimum of 8 credits, and as many as 12, in Final Project Final Project (CIS 609).

Under the supervision of a faculty member, the project may entail a group effort involving several master's degree students.

Accelerated Master's Degree Program

This program is open to students who earn a BS or BA degree in computer and information science at the University of Oregon and who want to enter the master's degree program.

If a UO undergraduate takes one or two 400-level electives that also are offered as 500-level courses, the student can petition the department to have 4 or 8 credits deducted from the total number of elective credits required for the master's degree. The student must earn an A– or better in the 400-level course and have an overall GPA of 3.50 in upper-division CIS courses to participate in this accelerated master's program. Note that all admission procedures, as outlined in the Master's Degree Program (p. 7) section, are also applicable. Applications are available in the department office.

Awards and Honor Societies

The Erwin and Gertrude Juilfs Scholarship in Computer and Information Science, in honor of Erwin and Gertrude Juilfs, is awarded to one or more students who show exceptional promise for achievement as evidenced by grade point average, originality of research, or other creative activities.

The J. Donald Hubbard Scholarship in Computer and Information Science, in honor of J. Donald Hubbard, recognizes an undergraduate or graduate student who shows outstanding promise in the fields of computer-human interaction, computer graphics, or multimedia.

The Gurdeep Pall Scholarship in Computer and Information Science, in honor of Gurdeep Pall, is awarded to a student based on the overall quality of their academic work, their commitment to learning, and their potential for further academic achievement.

Students with outstanding academic accomplishments may be invited to become members of Upsilon Pi Epsilon, the international honor society in computer science.

Doctoral Degree Program

The doctor of philosophy in computer and information science is, above all, a high-quality degree that is not conferred simply for the successful completion of a specified number of courses or years of study. It is a degree reserved for students who demonstrate a comprehensive understanding of computer science and an ability to do creative research. Each PhD student produces a significant piece of original research, presented in a written dissertation and defended in an oral examination.

The PhD program is structured to facilitate the process of learning how to do research. Students begin by taking required courses to build a foundation of knowledge that is essential for advanced research. Early in the program the student gains research experience by undertaking a directed research project under the close supervision of a faculty member and the scrutiny of a faculty committee. In the later stages of the program, students take fewer courses and spend most of their time exploring their dissertation area to learn how to identify and solve open problems. The final steps are to propose an independent research project, do the research, and write and defend a dissertation.

Admission

Application materials should be submitted by January 15 for the following fall term. Materials include everything required for admission to the master's program as well as a discussion of the anticipated research area.

Students who enter the UO with a master's degree may petition the Graduate Education Committee for credit toward the course requirements listed below, indicating how their prior graduate work corresponds to these courses. See the graduate coordinator for the petition.

PhD Degree Requirements

Core Courses [1]		
CIS 621	Algorithms and Complexity	4
CIS 624	Structure of Programming Languages	4
CIS 630	Distributed Systems	4
Cluster Courses		
Required course [2]		4
Two depth courses from cluster [2]		8
Electives		
Twelve credits of 500-level courses (including 510 courses) [3]		12
Twelve credits of 600-level courses [3]		12
Total Credits		**48**

1. Courses must be taken by the end of the student's second year.
2. Selected from a list of approved clusters. This list is updated each year to reflect experimental and other courses offered that year.
3. For graduate-level courses taken in other UO departments, a petition to the Graduate Education Committee is required.

PhD candidates who enter the program without a master's degree in computer science must take 48 credits in graduate course work including the core and cluster courses required for the MS program. Doctoral students must earn a minimum grade of B– and an overall GPA of 3.50 in the six courses they use to satisfy the core and cluster requirements.

Minimum Annual Enrollment

PhD students are expected to enroll in at least 6 credits of 600-level course work each year until their advancement to candidacy. Research: [Topic] (CIS 601), Dissertation (CIS 603), and Reading and Conference: [Topic] (CIS 605) do not satisfy this requirement. After candidacy, PhD students are encouraged to continue participation in 600-level courses

Directed Research Project

Complete a directed research project, which is supervised by a faculty member and evaluated by a faculty committee. The research project comprises the following:

1. The definition and expected results of the project in the form of a Directed Research Project Contract
2. Delivery of the materials constituting the results of the project and oral presentation of the results
3. A private oral examination by the committee members

Status Change

PhD candidates are admitted conditionally. Successful completion of the directed research project leads to a change in the student's doctoral status from conditional to unconditional.

Dissertation Advisory Committee

After successfully completing the directed research project, PhD students form a Dissertation Advisory Committee chaired by their research advisor. The main role of the committee is to advise the student between completion of the research project and mounting the dissertation defense. The committee takes primary responsibility for evaluating student progress. In addition, it approves the plan for the oral comprehensive examination, which in turn is approved by the graduate education committee. See the graduate coordinator for further instructions.

Area Examination

The student chooses an area of research and works closely with an advisor to learn the area in depth by surveying the current research and learning research methods, significant achievements, and how to pose and solve problems. The student gradually assumes a more independent role and prepares for the area examination, which tests depth of knowledge in the research area. The examination contains the following:

1. A survey of the area in the form of a position paper and an annotated bibliography
2. A public presentation of the position paper
3. A private oral examination by committee members

Advancement to Candidacy

After the area examination, the committee decides whether the student is ready for independent research work; if so, the student is advanced to candidacy.

Dissertation and Defense

Identify a significant unsolved research problem and submit a written dissertation proposal to the dissertation committee. The dissertation committee, comprising three department members and one member from an outside department, is approved by the graduate education committee. In addition to these four, the dissertation committee often includes a fifth examiner. This outside examiner should be a leading researcher in the candidate's field who is not at the University of Oregon. The outside member should be selected a year before the candidate's dissertation defense, and no later than six months before.

The student submits a written dissertation proposal to the committee for approval, and the proposal is then submitted to the graduate education committee. The proposal presents the research problems to be tackled, related research, methodology, anticipated results, and work plan. The committee may request an oral presentation, similar to the area exam, which allows the student to explain and answer question about the proposed research. The student then carries out the research.

The final stage is writing a dissertation and defending it in a public forum by presenting the research and answering questions about the methods and results. The dissertation committee may accept the dissertation, request small changes, or require the student to make substantial changes and schedule another defense

Graduate School Requirements

PhD students must meet the requirements set by the Graduate School as listed in that section of this catalog

Research Areas

It is important that a PhD student be able to work effectively with at least one dissertation advisor. Hence the student should identify, at an early stage, one or more areas of research to pursue. The student should also find a faculty member with similar interests to supervise the dissertation.

Courses

CIS 105. Explorations in Computing. 4 Credits.
Overview of basic ideas and areas of computer science: includes algorithms, hardware, machine organization, programming languages, networks, artificial intelligence, and associated ethical issues.

CIS 110. Fluency with Information Technology. 4 Credits.
Introduction to information technology (IT), the study of computer-based information systems. Basics of the Internet and World Wide Web. Students create websites using XHTML and CSS.

CIS 111. Introduction to Web Programming. 4 Credits.
Project-based approach to learning computer programming by building interactive web pages using JavaScript and XHTML. Programming concepts including structured and object-oriented program design. CIS 110 recommended preparation.

CIS 115. Multimedia Web Programming. 4 Credits.
Intermediate web programming with an emphasis on HTML5 multimedia: two-dimensional graphics, image processing, animation, video, user interaction, geolocation. Continuing JavaScript, DOM, Ajax, and JSON use, programming fundamentals, and debugging techniques.
Prereq: CIS 111.

CIS 122. Introduction to Programming and Problem Solving. 4 Credits.
Computational problem solving, algorithm design, data structures, and programming using a multi-paradigm programming language. Introduces techniques for program design, testing, and debugging.

CIS 196. Field Studies: [Topic]. 1-2 Credits.
Repeatable.

CIS 198. Workshop: [Topic]. 1-2 Credits.
Repeatable.

CIS 199. Special Studies in Computer Science: [Topic]. 5 Credits.
Repeatable.

CIS 210. Computer Science I. 4 Credits.
Basic concepts and practices of computer science. Topics include algorithmic problem solving, levels of abstraction, object-oriented design and programming, software organization, analysis of algorithm and data structures. Sequence.
Prereq: programming experience and MATH 112.

CIS 211. Computer Science II. 4 Credits.
Basic concepts and practices of computer science. Topics include algorithmic problem solving, levels of abstraction, object-oriented design and programming, software organization, analysis of algorithm and data structures. Sequence.
Prereq: CIS 210.

CIS 212. Computer Science III. 4 Credits.
Basic concepts and practices of computer science. Topics include algorithmic problem solving, levels of abstraction, object-oriented design and programming, software organization, analysis of algorithm and data structures. Sequence.
Prereq: CIS 211.

CIS 313. Intermediate Data Structures. 4 Credits.
Design and analysis of data structures as means of engineering efficient software; attention to data abstraction and encapsulation. Lists, trees, heaps, stacks, queues, dictionaries, priority queues.
Prereq: CIS 212, MATH 232 with grades of B- or better.

CIS 314. Computer Organization. 4 Credits.
Introduction to computer organization and instruction-set architecture--digital logic design, binary arithmetic, design of central processing unit and memory, machine-level programming.
Prereq: CIS 212, MATH 231 with grades of B- or better.

CIS 315. Intermediate Algorithms. 4 Credits.
Algorithm design, worst-case and average-behavior analysis, correctness, computational complexity.
Prereq: CIS 313.

CIS 322. Introduction to Software Engineering. 4 Credits.
A project-intensive introduction to software engineering intended to build skills, knowledge, and habits of mind that prepare students for 400-level computer science courses, internships, and other software.
Prereq: CIS 212.

CIS 330. C/C++ and Unix. 4 Credits.
Practical software design and programming activities in a C/C++ and Unix environment, with emphasis on the details of C/C++ and good programming style and practices.
Prereq: CIS 314.

CIS 399. Special Studies: [Topic]. 5 Credits.
Repeatable.

CIS 401. Research: [Topic]. 1-21 Credits.
Repeatable.

CIS 403. Thesis. 1-12 Credits.
Repeatable.

CIS 404. Internship; [Topic]. 1-4 Credits.
Repeatable.
Prereq: CIS 313.

CIS 405. Reading and Conference: [Topic]. 1-12 Credits.
Repeatable.

CIS 406. Field Studies: [Topic]. 1-21 Credits.
Repeatable.

CIS 407. Seminar: [Topic]. 1-5 Credits.
Repeatable. Opportunity to study in greater depth specific topics arising out of other courses.

CIS 408. Workshop: [Topic]. 1-21 Credits.
Repeatable.

CIS 409. Practicum: [Topic]. 1-2 Credits.
The student assists other students who are enrolled in introductory programming classes. For each four hours of scheduled weekly consulting, the student is awarded 1 credit. Repeatable for maximum of 4 credits.

CIS 410. Experimental Course: [Topic]. 5 Credits.
Repeatable.

CIS 413. Advanced Data Structures. 4 Credits.
Complex structures, storage management, sorting and searching, hashing, storage of texts, and information compression.
Prereq: CIS 315.

CIS 415. Operating Systems. 4 Credits.
Principles of operating system design. Process and memory management, concurrency, scheduling, input-output and file systems, security.
Prereq: CIS 330.

CIS 420. Automata Theory. 4 Credits.
Provides a mathematical basis for computability and complexity. Models of computation, formal languages, Turing machines, solvability. Nondeterminism and complexity classes.
Prereq: CIS 315.

CIS 422. Software Methodology I. 4 Credits.
Technical and nontechnical aspects of software development, including specification, planning, design, development, management and maintenance of software projects. Student teams complete projects.
Prereq: CIS 313.

CIS 423. Software Methodology II. 4 Credits.
Application of concepts and methodologies covered in CIS 422/522. Student teams complete a large system design and programming project. Final system specification, test plan, user documentation, and system walk throughs.
Prereq: CIS 422 with a B- or better.

CIS 425. Principles of Programming Languages. 4 Credits.
Syntax and semantics. Scope rules, environments, stores, denoted and expressed values, procedures, and parameters. Definitional interpreters. Types, overloading, parametric polymorphism, and inheritance. Varieties of abstraction.
Prereq: CIS 315.

CIS 427. Introduction to Logic. 4 Credits.
Basic notions of logic: propositional logic, first-order logic, Hilbert systems, sequent calculus, natural deduction. Soundness, completeness, undecidability. Current research in logic frameworks, automated deduction, Curry-Howard isomorphism.
Prereq: MATH 233; CIS 425 recommended pre or co-req.

CIS 429. Computer Architecture. 4 Credits.
RISC (reduced instruction-set computer) and CISC (complex instruction-set computer) design, storage hierarchies, high-performance processor design, pipelining, vector processing, networks, performance analysis.
Prereq: CIS 330.

CIS 431. Introduction to Parallel Computing. 4 Credits.
Parallel architecture, theory, algorithms, and programming with emphasis on parallel programming, focusing on models, languages, libraries, and runtime systems.
Prereq: CIS 330.

CIS 432. Introduction to Networks. 4 Credits.
Principles of computer network design. Link technologies, packet switching, routing, inter-networking, reliability. Internet protocols. Programming assignments focus on protocol design.
Prereq: CIS 330. CIS 415 recommended.

CIS 433. Computer and Network Security. 4 Credits.
Security for various aspects of computers and networks. Elementary cryptography, program security, trusted operating systems, network security, privacy, and legal and ethical issues.
Prereq: CIS 415.

CIS 441. Introduction to Computer Graphics. 4 Credits.
Introduction to the hardware, geometrical transforms, interaction techniques, and shape representation schemes that are important in interactive computer graphics. Programming assignments using contemporary graphics hardware and software systems.
Prereq: CIS 330.

CIS 443. User Interfaces. 4 Credits.
Introduction to user interface software engineering. Emphasis on theory of interface design, understanding the behavior of the user, and implementing programs on advanced systems.
Prereq: CIS 313.

CIS 445. Modeling and Simulation. 4 Credits.
Theoretical foundations and practical problems for the modeling and computer simulation of discrete and continuous systems. Simulation languages, empirical validation, applications in computer science.
Prereq: CIS 315, 330.

CIS 451. Database Processing. 4 Credits.
Fundamental concepts of DBMS. Data modeling, relational models and normal forms. File organization and index structures. SQL, embedded SQL, and concurrency control.
Prereq: CIS 313, 314.

CIS 452. Database Issues. 4 Credits.
Covers central database issues such as access methods, security, tuning, and concurrency control. Examines alternative database models.
Prereq: CIS 451.

CIS 453. Data Mining. 4 Credits.
Databases, machine learning, artificial intelligence, statistics, and data visualization. Examines data warehouses, data preprocessing, association and classification rule mining, and cluster analysis.
Prereq: CIS 451/551.

CIS 454. Bioinformatics. 4 Credits.
Introduction to bioinformatics from a computer science perspective covering algorithms for basic operations such as sequence comparison and phylogenetic inference on existing databases.

CIS 461. Introduction to Compilers. 4 Credits.
Lexical analysis, parsing, attribution, code generation.
Prereq: CIS 314, 425. CIS 420 strongly recommended.

CIS 471. Introduction to Artificial Intelligence. 4 Credits.
Basic themes, issues, and techniques of artificial intelligence, including agent architecture, knowledge representation and reasoning, problem solving and planning, game playing, and learning.
Prereq: CIS 315.

CIS 472. Machine Learning. 4 Credits.
A broad introduction to machine learning and its established algorithms. Topics include concept learning, decision trees, neural network.
Prereq: CIS 315.

CIS 473. Probabilistic Methods for Artificial Intelligence. 4 Credits.
Fundamental techniques for representing problems as probability distributions, performing inference, and learning from data. Topics include Bayesian and Markov networks, variable elimination, loopy belief propagation, and parameter.
Prereq: CIS 315.

CIS 490. Computer Ethics. 4 Credits.
Addresses ethical issues and social impacts of computing. Topics include crime, hacking, intellectual property, privacy, software reliability, employment, and worldwide networks.

CIS 503. Thesis. 1-16 Credits.
Repeatable.

CIS 507. Seminar: [Topic]. 1-5 Credits.
Repeatable. Opportunity to study in greater depth specific topics arising out of other courses.

CIS 508. Workshop: [Topic]. 1-21 Credits.
Repeatable.

CIS 510. Experimental Course: [Topic]. 5 Credits.
Repeatable.

CIS 513. Advanced Data Structures. 4 Credits.
Complex structures, storage management, sorting and searching, hashing, storage of texts, and information compression.

CIS 520. Automata Theory. 4 Credits.
Provides a mathematical basis for computability and complexity. Models of computation, formal languages, Turing machines, solvability. Nondeterminism and complexity classes.

CIS 522. Software Methodology I. 4 Credits.
Technical and nontechnical aspects of software development, including specification, planning, design, development, management and maintenance of software projects. Student teams complete projects.

CIS 523. Software Methodology II. 4 Credits.
Student teams complete a large system design and programming project. Final system specifications, test plan, user documentation, and system walk-through.
Prereq: CIS 522

CIS 527. Introduction to Logic. 4 Credits.
Basic notions of logic: propositional logic, first-order logic, Hilbert systems, sequent calculus, natural deduction. Soundness, completeness, undecidability. Current research in logic frameworks, automated deduction, Curry-Howard isomorphism.

CIS 529. Computer Architecture. 4 Credits.
RISC (reduced instruction-set computer) and CISC (complex instruction-set computer) design, storage hierarchies, high-performance processor design, pipelining, vector processing, networks, performance analysis.

CIS 531. Introduction to Parallel Computing. 4 Credits.
Parallel architecture, theory, algorithms, and programming with emphasis on parallel programming, focusing on models, languages, libraries, and runtime systems.

CIS 532. Introduction to Networks. 4 Credits.
Principles of computer network design. Link technologies, packet switching, routing, inter-networking, reliability. Internet protocols. Programming assignments focus on protocol design.

CIS 533. Computer and Network Security. 4 Credits.
Security for various aspects of computers and networks. Elementary cryptography,program security, trusted operating systems, network security, privacy, and legal and ethical issues.

CIS 541. Introduction to Computer Graphics. 4 Credits.
Introduction to the hardware, geometrical transforms, interaction techniques, and shape representation schemes that are important in interactive computer graphics. Programming assignments using contemporary graphics hardware and software systems.

CIS 543. User Interfaces. 4 Credits.
Introduction to user interface software engineering. Emphasis on theory of interface design, understanding the behavior of the user, and implementing programs on advanced systems.

CIS 545. Modeling and Simulation. 4 Credits.
Theoretical foundations and practical problems for the modeling and computer simulation of discrete and continuous systems. Simulation languages, empirical validation, applications in computer science.

CIS 551. Database Processing. 4 Credits.
Fundamental concepts of DBMS. Data modeling, relational models and normal forms. File organization and index structures. SQL, embedded SQL, and concurrency control.

CIS 552. Database Issues. 4 Credits.
Covers central database issues such as access methods, security, tuning, and concurrency control. Examines alternative database models.
Prereq: CIS 4/551.

CIS 553. Data Mining. 4 Credits.
Databases, machine learning, artificial intelligence, statistics, and data visualization. Examines data warehouses, data preprocessing, association and classification rule mining, and cluster analysis.
Prereq: CIS 451/551.

CIS 554. Bioinformatics. 4 Credits.
Introduction to bioinformatics from a computer science perspective covering algorithms for basic operations such as sequence comparison and phylogenetic inference on existing databases.

CIS 561. Introduction to Compilers. 4 Credits.
Lexical analysis, parsing, attribution, code generation.
Prereq: CIS 314 or equivalent, 624. CIS 420/520 strongly recommended.

CIS 571. Introduction to Artificial Intelligence. 4 Credits.
Basic themes, issues, and techniques of artificial intelligence, including agent architecture, knowledge representation and reasoning, problem solving and planning, game playing, and learning.

CIS 572. Machine Learning. 4 Credits.
A broad introduction to machine learning and its established algorithms. Topics include concept learning, decision trees, neural network.

CIS 573. Probabilistic Methods for Artificial Intelligence. 4 Credits.
Fundamental techniques for representing problems as probability distributions, performing inference, and learning from data. Topics include Bayesian and Markov networks, variable elimination, loopy belief propagation, and parameter.

CIS 590. Computer Ethics. 4 Credits.
Addresses ethical issues and social impacts of computing. Topics include crime, hacking, intellectual property, privacy, software reliability, employment, and worldwide networks.

CIS 601. Research: [Topic]. 1-16 Credits.
Repeatable.

CIS 602. Supervised College Teaching. 1-5 Credits.
Repeatable.

CIS 603. Dissertation. 1-16 Credits.
Repeatable.

CIS 604. Internship: [Topic]. 1-4 Credits.
Repeatable.

CIS 605. Reading and Conference: [Topic]. 1-16 Credits.
Repeatable.

CIS 606. Field Studies: [Topic]. 1-16 Credits.
Repeatable.

CIS 607. Seminar: [Topic]. 1-5 Credits.
Repeatable. Research topics are presented.

CIS 608. Colloquium: [Topic]. 1 Credit.
Repeatable.

CIS 609. Final Project. 1-16 Credits.
Repeatable. Final project for master's degree without thesis.

CIS 610. Experimental Course: [Topic]. 1-5 Credits.
Repeatable.

CIS 621. Algorithms and Complexity. 4 Credits.
Design and analysis of algorithms, strategies for efficient algorithms, introduction to complexity theory including NP-completeness.
Prereq: CIS 420/520 strongly recommended.

CIS 622. Theoretical Foundations: [Topic]. 4 Credits.
Selected topics from computability and complexity theory. Repeatable twice when topic changes for maximum of 12 credits.
Prereq: CIS 621.

CIS 624. Structure of Programming Languages. 4 Credits.
Introduction to axiomatic, operational, and denotational semantics. Environments, stores, and continuations. Type theory, subtypes, polymorphism, and inheritance. Functional and logic programming.

CIS 630. Distributed Systems. 4 Credits.
Principles of distributed computer systems: interprocess communication, distributed file systems, distributed timing and synchronization, distributed programming, transactions, process scheduling, distributed shared memory.
Prereq: CIS 415 or equivalent, CIS 429/529.

CIS 631. Parallel Processing. 4 Credits.
Advanced topics in parallel processing including massively parallel computer architecture, supercomputers, parallelizing compiler technology, performance evaluation, parallel programming languages, parallel applications.
Prereq: CIS 429/529.

CIS 632. Computer Networks. 4 Credits.
Advanced issues in computer networks, focusing on research to extend the services offered by the Internet.
Prereq: CIS 432/532.

CIS 633. Advanced Network Security. 4 Credits.
Classic and state-of-the-art research topics in network security; threats and attacks, defense algorithms and mechanisms, measurement and evaluation of both security problems and solutions. Offered alternate years.
Prereq: CIS 533.

CIS 640. Writing in Computer Research. 2 Credits.
Students learn to provide and accept constructive criticism of writing samples in a workshop format.

CIS 650. Software Engineering. 4 Credits.
Examines recent models and tools in software engineering including modifications to the traditional software life-cycle model, development environments, and speculative view of the future role of artificial intelligence.

Courses

CIT 281. Web Applications Development I. 4 Credits.
Fundamentals of web application development using open-source software tools and technologies (Unix, Git), client-side frameworks, server-side programming (Node.js, PHP), model-view-controller pattern, data storage and APIs, cloud hosting.
Prereq: CIS 111 with a B- or higher.

CIT 381. Database Systems. 4 Credits.
Introduction to database systems, emphasis on database design and access. Database concepts, data modeling, SQL, connecting database to web.
Prereq: CIT 281.

CIT 382. Web Applications Development II. 4 Credits.
Server- and client-side technologies and their interaction for database-driven web applications: application frameworks, single-page applications, cloud platforms, and open-source software stacks—MEAN (MongoDB, ExpressJS, AngularJS, Node.js) versus LAMP (Linux, Apache, MySQL, PHP).
Prereq: CIT 381.

CIT 383. Networking Fundamentals. 4 Credits.
Fundamentals of data communication and networks. Network management and security.
Prereq: CIT 382.

CIT 405. Reading and Conference: [Topic]. 1-4 Credits.
Repeatable.