Computer Science

Degrees Offered

- Master of Science (Computer Science)
- Doctor of Philosophy (Computer Science)
- Professional Master's Degree - Online
- Graduate Certificate in Cybersecurity

Program Overview

The Computer Science Department offers the Master of Science and Doctor of Philosophy in Computer Science. These degree programs demand academic rigor and depth yet also address real-world problems.

The department’s research falls into six areas: algorithms, human-centered computing, machine learning and data science, robotics, security and privacy. The Department of Computer Science has a strong emphasis on research, both within the department and collaboratively with other departments, universities, government organizations and industry partners. See the CS@Mines Research webpage for updated details and current projects: https://cs.mines.edu/research/.

PROGRAM DETAILS

The Computer Science Department offers a variety of programs:

We offer the degrees Master of Science and Doctor of Philosophy in Computer Science. The master's program is designed to prepare candidates for careers in industry or government or for further study at the PhD level; both thesis and non-thesis options are available. The PhD degree program is sufficiently flexible to prepare candidates for careers in industry, government, or academia.

Combined Program: The CS Department also offers combined BS/MS degree programs. These programs offer an expedited graduate school application process and allow students to begin graduate coursework while still finishing their undergraduate degree requirements.

Admission PREREQUISITES

BS+MS Combined

Current Mines undergraduate students are encouraged to apply for the combined program once they have taken five or more Computer Science classes at Mines (classes transferred from other universities will not be considered). This requirement may be met by any 200-level or above course with a CSCI prefix (e.g., CSCI200, CSCI306, CSCI406 CSCI442, etc.), excluding CSCI274, CSCI370, CSCI499.

Students should have an overall GPA of at least 3.0 and a GPA of 3.2 for courses in the major. The calculation of GPA in the major will be based on *all* 200-level or above CSCI courses except those excluded above (i.e., CSCI274, CSCI370 and CSCI499). If a course is taken multiple times, all of the grades will be included into the GPA calculation. Interested students with a lower GPA must include in their statement of goals/personal statement a section explaining why they should be admitted to the program.

GRADUATE CERTIFICATE IN CYBERSECURITY

The minimum requirements for admission to the Graduate Certificate in CyberSecurity are:

- Undergraduate-level knowledge on data structures, computer organization, algorithms, and operating systems.
- Undergraduate-level knowledge on statistics and discrete mathematics.
- Undergraduate-level skills on the Linux operating system and shell scripts.
- Undergraduate-level programming skills in languages such as C, C++, Python, Java, JavaScript, and HTML/CSS.

MASTER'S AND PHD

The minimum requirements for admission to the MS and Ph.D degrees in Computer Science are:

- Applicants must have a bachelor's degree, or equivalent, from an accredited institution with a grade-point average of 3.0 or better on a 4.0 scale.
- Students are expected to have completed the following coursework at Mines or equivalent at another institution: 1) CSCI200 FOUNDATIONAL PROGRAMMING CONCEPTS & DESIGN: Foundational Programming Concepts and Design, 2) CSCI220: Data Structures and Algorithms, 3) CSCI306: Software Engineering, 4) CSCI341: Computer Organization, 5) CSCI358: Discrete Mathematics. It is strongly suggested applicants also have completed a course in Algorithms and Operating Systems. The CS graduate admissions committee will require that students who do not meet this expectation demonstrate competency or take remedial coursework. Such coursework will not count toward the graduate degree. The committee will decide whether to recommend regular or provisional admission.
- TOEFL score of 79 or higher (or 550 for the paper-based test or 213 for the computer-based test) for applicants whose native language is not English. In lieu of a TOEFL score, and IELTS score of 6.5 or higher will be accepted.
- For the PhD program, prior research experience is desired but not required.

Admitted Students: The CS Department Graduate Committee may require that an admitted student take undergraduate remedial coursework to overcome technical deficiencies. The committee will decide whether to recommend regular or provisional admission.

Transfer Courses: Graduate-level courses taken at other universities for which a grade equivalent to a B or better was received will be considered for transfer credit with approval of the advisor and/or thesis committee, and CS department head, as appropriate. Transfer credits must not have been used as credit toward a Bachelor degree. For the MS degree, no more than nine credits may transfer. For the PhD degree, up to 24 credits may be transferred. In lieu of transfer credit for individual courses, students who enter the PhD program with a thesis-based master's degree from another institution may transfer up to 36 credits in recognition of the coursework and research completed for that degree.

Advisor and Thesis Committee: Students must have an advisor from the CS faculty to direct and monitor their academic plan, research, and independent studies. Advisors must be full-time permanent members of the faculty. In this context, full-time permanent members of the faculty are those that hold the rank of professor, associate professor, assistant professor, research professor, associate research professor or assistant research professor. Upon approval by the graduate dean, adjunct faculty, teaching faculty, visiting professors, emeritus professors, and off-campus representatives may be designated additional co-advisors. A list of CS faculty by rank is available in the faculty tab of the catalog.

Master of Science (thesis option) students in CS must have at least three members on their thesis committee; the advisor and one other member must be permanent faculty in the CS Department. CS PhD thesis committees must have at least four members; the advisor/co-advisor and two additional members must be permanent faculty in the CS Department, and one member must be outside the departmental faculty and serving as chair of the committee. Students who choose to have a minor program must select a representative from the minor area of study to serve on the thesis committee.

Degree Audit and Admission to Candidacy: Master's students must complete the Degree Audit form by the posted deadline. PhD students need to submit the Degree Audit form by the posted deadline and need to submit the Admission to Candidacy form by the posted deadline of the semester in which they want to be considered eligible for reduced registration.

Time Limit: As stipulated by the Mines Graduate School, a candidate for a master's degree must complete all requirements for the degree within five years of the date of admission into the degree program. A candidate for a doctoral degree must complete all requirements for the degree within nine years of the date of admission into the degree program.

PROGRAM REQUIREMENTS

Mines' Combined Undergraduate/Graduate Degree Program

Students enrolled in Mines' combined undergraduate/graduate program may double count up to 6 credits of graduate coursework (500-level or above) to fulfill requirements of both their undergraduate and graduate degree programs. These courses must have been passed with B- or better, not be substitutes for required coursework, and meet all other university, department, and program requirements for graduate credit.

Students are advised to consult with their undergraduate and graduate advisors for appropriate courses to double count upon admission to the combined program.

Current Mines undergraduate students are encouraged to apply for the combined program once they have taken five or more Computer Science classes at Mines (classes transferred from other universities will not be considered). This requirement may be met by any 200-level or above course with a CSCI prefix (e.g., CSCI200, CSCI306, CSCI406, CSCI442, etc.), excluding CSCI274, CSCI370, and CSCI499. Students should have an overall GPA of at least 3.0 and a GPA of 3.2 for courses in the major. The calculation of GPA in the major will be based on all 200-level or above CSCI courses except those excluded above (i.e., CSCI274, CSCI370 and CSCI499). If a course is taken multiple times, all of the grades will be included into the GPA calculation. Interested students with a lower GPA must include in their statement of goals/personal statement a section explaining why they should be admitted to the program.

Master of Science - Computer Science

The MS degree in Computer Science (thesis or non-thesis option) requires 30 credits. Requirements for the thesis MS are 21 hours of coursework plus 9 hours of thesis credit leading to an acceptable master's thesis; thesis students are encouraged to find a thesis advisor and form a thesis committee by the end of the first year. The non-thesis option consists of two tracks: a Project Track and a Coursework Track. Requirements for the Project Track are 24 hours of coursework plus 6 hours of project credit; requirements for the Coursework Track are 30 hours of coursework. The following two core courses are required of all students. Students may choose elective courses from any CSCI graduate course offered by the department. In addition, up to 6 credits of elective courses may be taken outside of CSCI. Lastly, a maximum of six independent study course units can be used to fulfill degree requirements. Core courses for the program are below. CSCI 561 and CSCI 564 (or an alternative course, subject to approval by the committee based on the provided justification). Examples of potential alternative courses include CSCI 582 (Computing Beyond CPUs), CSCI 565 (Distributed Computing), and CSCI 563 (Parallel Computing for Scientists and Engineers)..

CSCI561	THEORY OF COMPUTATION (offered every fall)	3.0
CSCI564	ADVANCED COMPUTER ARCHITECTURE (offered every spring)	3.0

MS Project Track

Students are required to take 6 credits of CSCI700 to fulfill the MS project requirement. (It is recommended that the 6 credits consist of two consecutive semesters of 3 credits each.) At most, 6 credits of CSCI700 will be counted toward the masters non-thesis degree. Deliverables include a report and a presentation to a committee of two CS faculty including the advisor. Deliverables must be successfully completed in the last semester in which the student registers for CSCI700. A student must receive two pass votes (i.e., a unanimous vote) to satisfy the project option.

MS Thesis Defense

At the conclusion of the MS thesis, the student will be required to make a formal presentation and defense of her/his thesis research. A student must pass this defense to earn an MS degree.

CS Minor

A CS Minor at the master's level requires a minimum of 9 credits of CSCI coursework, of which at least 6 credits of coursework must be at the 500-level or above excluding independent studies and graduate seminars. Pursuant to Graduate School rules, all minors must be approved by the student's advisor, home department head, and a faculty representative of the minor area of study. A minor may not be taken in the student's major area of study.

CS@Mines Bridge Program

The Bridge Program is designed for those without a background in CS to have a direct pathway to a Computer Science (CS) master's (MS) degree. Students in the CS@Mines Bridge program will complete foundational undergraduate courses which will prepare them for master's level coursework. Students will then continue to the MS non-thesis (NT) program as a full-time or part-time student.

All incoming Bridge students are required to have earned a bachelor's degree from an accredited with a grade-point average of at least 3.0 on a 4.0 scale.

Bridge students are required to take a set of pre-requisite Bridge courses to prepare for the MS degree. Required courses are CSCI195, CSCI200, CSCI210, CSCI220, CSCI306, CSCI341, and CSCI358. Incoming students with no completed coursework in calculus or programming from an accredited institution may need to take Calculus 1-3 and CSCI128 in addition to the Bridge courses stated above. Students must earn a B or higher in their Bridge coursework for it to count toward their progress in the program.

For more information regarding the program and application process, please contact our Graduate Program Manager, Kelsie Diaz, at kdiaz@mines.edu.

Doctor of Philosophy - Computer Science

The PhD degree in Computer Science requires 72 credits of coursework and research credits. Required coursework provides a strong background in computer science. A course of study leading to the PhD degree can be designed either for the student who has completed the master's degree or for the student who has completed the bachelor's degree. The following three courses are required of all students. Students who have taken equivalent courses at another institution may satisfy these requirements by transfer. Core courses for the program are below. CSCI 561 and CSCI 564 (or an alternative course, subject to approval by the committee based on the provided justification). Examples of potential alternative courses include CSCI 582 (Computing Beyond CPUs), CSCI 565 (Distributed Computing), and CSCI 563 (Parallel Computing for Scientists and Engineers).

CSCI561	THEORY OF COMPUTATION (offered every fall)	3.0
CSCI564	ADVANCED COMPUTER ARCHITECTURE (offered every spring)	3.0
SYGN502	INTRODUCTION TO RESEARCH ETHICS	1.0

PhD Qualifying Examination

Students desiring to take the PhD qualifying exam must have:

- If required by your advisor, taken SYGN501: The Art of Science (previously or concurrently).
- Completed (previously or concurrently) all core courses.
- Complete (previously or concurrently) at least four CSCI 500-level courses at Mines (only one CSCI599 is allowed).
- Maintained a GPA of 3.5 or higher in all CSCI 500-level courses taken.

The PhD qualifying exam must be passed no later than the fourth semester of study. Exception must be formally requested via email to the qualifying exam committee chair and approved by the graduate committee. The PhD qualifying exam is offered once a semester. Each PhD qualifying exam comprises of two research areas, chosen by the student. The exam consists of the following steps:

Step 1. A student indicates intention to take the CS PhD qualifying exam by choosing two research interest areas.

- The primary test area should be the same as the research area of the student's (potential) advisor. This exam will be more open-ended research than the second test area. A formal written report and a formal presentation meeting are required for this exam. The outcome of this exam can be part of the student's dissertation research. In fact, the student is strongly encouraged to create results that can lead to a publication. It is acceptable and encouraged if the advisor is involved to provide suggestions. The student is required to clearly document in the written report how the advisor was involved in the exam.
- The second test area should be from another research area of interest to the student that is 1) supported by faculty within the CS department and 2) different from the student's primary advisor's research area. It is highly recommended that the student choose their secondary test area with an instructor the student has had in one or more courses. This exam will likely be less substantial than the primary exam, i.e., instructions will be more concrete. The purpose of having a second test area is to ensure students can demonstrate both the breadth of knowledge and the capability in doing independent research. Thus, no faculty member is allowed to assist the student in this second exam except for answering clarification questions.

Students must inform the CS Qualifying Committee Chair of their intention to take the exam by a specific date provided by the Chair in the prior semester.

Step 2. The qualifying exam committee chair creates an exam committee of (at least) four appropriate faculty. The exam committee assigns the student specific tasks with corresponding deliverables for both research areas chosen. The tasks will be some combination from the following list:

- design and evaluate new algorithms or systems for an important research problem, and write a report that summarizes the design and the evaluation results.
- read a set of technical papers, write a summary of the papers read, make a presentation, and answer questions (presentations will be limited to 30-minutes with a hard stop not including Q&A).
- complete a hands-on activity (e.g., develop research software) and write a report that explains the difficulties with the activity and what was learned.
- complete a set of take-home problems.
- write a literature survey (i.e., track down references, separate relevant from irrelevant papers).

Step 3. The student must complete all deliverables no later than the Monday of review week (11:59 p.m.). Failure to meet the deadline is considered a failed attempt. The submitted report on the deadline is considered to be final, i.e., no update is allowed after the due date/time. The oral presentation will be scheduled during Finals Week. Before the oral presentation, the student is not allowed to practice the exam presentation with his/her advisor or research group to get feedback. The student will access exam problems, and submit deliverables through a specified system such as Canvas course/module. Additionally, the specified system will be used to deliver feedback from the committee to the student outlining strengths, weaknesses, recommendations and exam results.

Step 4. Each member of the exam committee makes a recommendation on the deliverables from the following list: strongly support, support, and do not support. To pass the PhD qualifying exam, the student must have at least two "strongly supports" and no more than one "do not support." If a student receives two or more "do not support" votes by the committee members, the student fails the exam. All other cases other than pass or fail are considered as conditional pass.

Conditional Pass Requirements

If a student receives a conditional pass, the student is required to take (an) additional test(s). The exam committee will explicitly specify the deadline for the student to take the additional test in the feedback comments to the student. The deadline will likely be in the beginning weeks of the following semester. The additional test(s) may be the whole or part(s) of the original qualifying exam or may be an additional task, as determined by the exam committee. If the student passes the assigned additional test, the conditional pass will be converted into a pass; otherwise, the outcome of the qualifying exam will be a fail.

The student is informed of the qualifying exam decision (pass, fail, or conditional pass) no later than the Monday after finals week. The student is informed of the outcome of a conditional pass test within two weeks after the test. A student can only fail the exam one time. If a second failure occurs, the student has unsatisfactory academic performance that results in an immediate, mandatory dismissal of the graduate student from the PhD program.

PhD Thesis Proposal: After passing the qualifying exam, the PhD student is allowed up to 18 months to prepare a written thesis proposal and present it formally to the student's thesis committee and other interested faculty.

Admission to Candidacy: In addition to the Graduate School requirements, full-time PhD students must complete the following requirements within two calendar years of enrolling in the PhD program.

- Have a thesis committee appointment form on file in the Graduate Office.
- Have passed the PhD qualifying exam demonstrating adequate preparation for, and satisfactory ability to conduct doctoral research.

PhD Thesis Defense: At the conclusion of the student's PhD program, the student will be required to make a formal presentation and defense of her/his thesis research. A student must pass this defense to earn a PhD degree.

CS Minor

A CS Minor at the PhD level requires a minimum of 12 credits of CSCI coursework, of which 9 credits of coursework must be at the 500-level or above excluding independent studies and graduate seminars. Pursuant to Graduate School rules all minors must be approved by the student's advisor, home department head, and a faculty representative of the minor area of study. A minor may not be taken in the student's major area of study.

PROFESSIONAL MASTERS DEGREE IN COMPUTER SCIENCE

To earn a professional master's (MP-CS) degree, students must complete 30 credits as outlined below. The program is composed of four flexible tracks from the options listed. To complete the 30 credit hours, students would be required to complete any 3 tracks, along with an additional elective course. The required elective course may be fulfilled by any CSCI course from the track options listed if it is not being used to complete one of their chosen tracks. At least 7 out of 10 required courses MUST be CSCI/DSCI. Students may enroll in the professional master's program from the outset and take courses from the various tracks in any order.

- All courses are delivered online; however, students interested in on-campus classes may substitute one course per track with a pre-approved CSCI elective listed in our Graduate Catalog.
- As we develop more courses for our professional master's program, we may decide that some of these courses are not appropriate for our in-residence graduate MS and PhD students and therefore may restrict registration to only MP-CS students.

Our plan is to have the Professional Master's program ready to launch in Fall 2025. We will increase the number of courses available for the MP-CS degree over the coming semesters to give both students and faculty more flexibility with offerings. Currently, the plan for our program for Fall 2025 will consist of the following courses and tracks (**bold** indicates course(s) to be developed over the next 9 months):

Track 1: Applied Machine Learning

The track in Applied Machine Learning is targeted to train recent graduates or mid-career professionals with a bachelor's degree in computer science or other STEM-related field with acquired basic knowledge in programming, data structures, statistics, and linear algebra.

To complete the track in Applied Machine Learning, students must complete 9 hours of coursework as follows:

- **Required Courses**
  - DSCI 503: Advanced Data Science
  - DSCI 570: Machine Learning
- **Pick one from the following:**
  - DSCI 530: Statistical Methods
  - DSCI 575: Advanced Machine Learning
  - **CSCI 578: Bioinformatics (Assignments are geared towards analytics *)**
  - **CSCI 507: Computer Vision (Assignments are geared towards analytics *)**

** Note: The assignments are focused on applying analytical techniques and methodologies, with an emphasis on data analysis, interpretation, and deriving insights from data. They are designed to develop students' skills in exploring, processing, and making data-driven decisions using various tools and frameworks commonly used in analytics.*

Track 2: Cybersecurity

The track in Cybersecurity is targeted to train recent graduates or mid-career professionals with a bachelor's degree in computer science or a related field with a foundational knowledge in programming, data structures, algorithms, systems programming, statistics, and discrete math.

To complete the track in Cybersecurity, students must complete 9 hours of coursework as follows:

- **Required Courses**
  - CSCI 574: Theory of Cryptography
  - CSCI 585: Information Security and Privacy

- **Pick one from the following:**
    - CSCI 560: Fundamentals of Computer Networks
    - CSCI 587: Cyber Physical Systems Security

Track 3: Analytics and Project Management

The track in Analytics and Project Management is targeted to train recent graduates or mid-career professionals with a bachelor's degree in computer science or other STEM-related field with acquired basic knowledge in programming, data structures, statistics, and linear algebra.

To complete the track in Analytics and Project Management, students must complete 9 hours of coursework as follows

- **Required Course:**
    - EBGN 525: Business Analytics

- **Pick one from the following:**
    - EBGN553: Project Management
    - EBGN 563: Management of Technology and Innovation (taught by Scott Houser)
    - EBGN 565: Marketing for Technology-based companies
    - EBGN 571: Marketing Analytics
    - EBGN 577: Leading and Managing High Performance Teams

Projects in these courses would focus on software development or data science/machine learning, with an emphasis on requiring more in-depth software development and analysis. As the online program expands, we plan to add more computer science-oriented courses to this certificate track, such as cybersecurity leadership and operations.

Track 4: #Advanced Software Technologies

The track in Advanced Software Technologies is targeted to train recent graduates or mid-career professionals with a bachelor's degree in computer science or other STEM-related field with acquired basic knowledge in programming, data structures, and software engineering.

To complete the track in Advanced Software Technologies, students must complete 9 hours of coursework as follows:

- **Required Course:**
    - **CSCI 5xx: Advance Software Techniques and Project Development**

- **Pick two from the following. Note that these courses would center around major software projects:**
    - **CSCI 578: Bioinformatics (Assignments are geared towards software design ^)**
    - **CSCI 507: Computer Vision (Assignments are geared towards software design ^)**

^ *Note: The assignments are focused on developing and applying software design principles, emphasizing the creation of robust, scalable, and efficient systems. They are designed to enhance students' skills in structuring, designing, and implementing software solutions, while considering best practices, design patterns, and user needs within real-world scenarios.*

Of the courses included in our four tracks, all except 'Advanced Software Techniques and Project Development' already exist in our in-person or online curriculum.

During the 2025-2026 academic year, we plan to develop additional courses in consultation with our external advisory committee to address the high priority needs of working professionals. Based on ongoing discussions, we envision offering the following courses to our MP-CS students.

Key: All courses will be [3.0 credit hours]. Courses with a 5xx are proposed and are in various stages of discussion for development (e.g., **bold**: priority courses, *Italic*: aspirational courses)

Track 1: Applied Machine Learning

The track in Applied Machine Learning is targeted to train recent graduates or mid-career professionals with a bachelor's degree in computer science or other STEM-related field with acquired basic knowledge in programming, data structures, statistics, and linear algebra.

To complete the track in Applied Machine Learning, students must complete 9 hours of coursework as follows:

- **Required Courses**
    - DSCI 503: Advanced Data Science
    - DSCI 570: Machine Learning

- **Pick one from the following:**
    - DSCI 530: Statistical Methods
    - DSCI 575: Advanced Machine Learning
    - **CSCI 578: Bioinformatics (Assignments are geared towards analytics *)**
    - **CSCI 507: Computer Vision (Assignments are geared towards analytics *)**
    - *CSCI 504: Artificial Intelligence*

* Note: The assignments are focused on applying analytical techniques and methodologies, with an emphasis on data analysis, interpretation, and deriving insights from data. They are designed to develop students' skills in exploring, processing, and making data-driven decisions using various tools and frameworks commonly used in analytics.

Track 2: Cybersecurity -

The track in Cybersecurity is targeted to train recent graduates or mid-career professionals with a bachelor’s degree in computer science or a related field with a foundational knowledge in programming, data structures, algorithms, systems programming, statistics, and discrete math.

To complete the track in Cybersecurity, students must complete 9 hours of coursework as follows:

- **Required Courses**
  - CSCI 574: Theory of Cryptography
  - CSCI 585: Information Security and Privacy
- **Pick one from the following:**
  - CSCI 560: Fundamentals of Computer Networks
  - CSCI 587: Cyber Physical Systems Security
  - *CSCI 5xx: Cybersecurity Leadership and Operations (this course is added to help prepare students for the CISSP exam from ISC2).*

Track 3: Analytics and Project Management

The track in Analytics and Project Management is targeted to train recent graduates or mid-career professionals with a bachelor's degree in computer science or other STEM-related field with acquired basic knowledge in programming, data structures, statistics, and linear algebra.

To complete the track in Analytics and Project Management, students must complete 9 hours of coursework as follows:

- **Required Course:**
  - EBGN 525: Business Analytics (python-based intro to DS)
- **Pick one from the following:**
  - EBGN 553: Project Management
  - EBGN 563: Management of Technology and Innovation (taught by Scott Houser)
  - EBGN 565: Marketing for Technology-based companies
  - EBGN 571: Marketing Analytics
  - EBGN 577: Leading and Managing High Performance Teams
  - *CSCI 5xx: Cybersecurity Leadership and Operation (see note above under track 2) The course has a business management aspect that could allow it to fit in this track as well as the Cybersecurity track)*
  - *EBGN 5xx: Innovation & Entrepreneurship in Engineering & Technology*

Projects in these courses would focus on software development or data science/machine learning, with an emphasis on requiring more in-depth software development and analysis. As the online program expands, we plan to add more computer science-oriented courses to this certificate track, such as cybersecurity leadership and operations.

Track 4: Advanced Software Technologies

The track in Advanced Software Technologies is targeted to train recent graduates or mid-career professionals with a bachelor's degree in computer science or other STEM-related field with acquired basic knowledge in programming, data structures, and software engineering.

To complete the track in Advanced Software Technologies, students must complete 9 hours of coursework as follows:

- **Required Course:**
  - **CSCI 5xx: Advance Software Techniques and Project Development**
- **Pick two from the following:**
  - CSCI 5xx: Verification, Testing, Quality Assurance
  - **CSCI 578: Bioinformatics (Assignments are geared towards software design ^)**
  - **CSCI 507: Computer Vision (Assignments are geared towards software design^)**
  - *CSCI 5xx: UI / UX*
  - *CSCI 5xx: Mobile App Development*
  - *CSCI 5xx: Cloud Computing*
  - *CSCI 5xx: Web Programming and Applications*

^ Note: The assignments are focused on developing and applying software design principles, emphasizing the creation of robust, scalable, and efficient systems. They are designed to enhance students' skills in structuring, designing, and implementing software solutions, while considering best practices, design patterns, and user needs within real-world scenarios.

Graduate Certificate in Cybersecurity

The online Graduate Certificate in Cybersecurity is targeted to train recent graduates or mid-career professionals with a bachelor’s degree in computer science or a related field with a foundational knowledge in programming, data structures, algorithms, systems programming, statistics, and discrete math.

To complete the online Graduate Certificate in Cybersecurity, students must complete 9 hours of coursework as follows:

- Required Courses:

  o CSCI 574: Theory of Cryptography

  o CSCI 585: Information Security and Privacy

- Pick one from the following:

  o CSCI 560: Fundamentals of Computer Networks

  o CSCI 587: Cyber Physical Systems Security

While all courses are to be delivered online, students that are interested in attending an on-campus class may substitute one of the above courses with a pre-approved elective as listed in our Graduate Catalog.

FUTURE PLANS

In the 2025-2026 academic year, we aim to expand our course offerings in collaboration with our external advisory committee to meet the evolving needs of working professionals. Based on our ongoing conversations, we plan to introduce a new course focused on Cybersecurity Leadership and Operations to address key industry priorities. This course, designed to support students in preparing for the CISSP exam from ISC2, will complement the following curriculum:

- Required Courses:

    o CSCI 574: Theory of Cryptography

    o CSCI 585: Information Security and Privacy

- Pick one from the following:

    o CSCI 560: Fundamentals of Computer Networks

    o CSCI 587: Cyber Physical Systems Security

    o CSCI 5xx: Cybersecurity Leadership and Operations (this course is added to help prepare students for the CISSP exam from ISC2).

Courses

CSCI500. GRADUATE SEMINAR. 0.0 Semester Hrs.
This zero-credit graduate course builds on the CS department seminars in the colloquium series, which consist of presentations delivered by external or internal invited speakers on topics broadly related to computer science. The seminar is mandatory for all Ph.D. students. The students are required to enroll in the course every semester. Any student who cannot take the course for valid reasons should notify their advisor, who will then make a request to the CS graduate committee for a waiver. These requests could be for the duration of one semester or longer. The course will be graded as PRG/PRU based on student attendance at the department seminars in the colloquium series - the student has to attend at least two thirds of all the seminars each semester in order to get a PRG grade.
Course Learning Outcomes

CSCI507. INTRODUCTION TO COMPUTER VISION. 3.0 Semester Hrs.
Equivalent with CSCI437,CSCI512,EENG507,EENG512,
(I) Computer vision is the process of using computers to acquire images, transform images, and extract symbolic descriptions from images. This course provides an introduction to this field, covering topics in image formation, feature extraction, location estimation, and object recognition. Design ability and hands-on projects will be emphasized, using popular software tools. The course will be of interest both to those who want to learn more about the subject and to those who just want to use computer imaging techniques. 3 hours lecture; 3 semester hours. Prerequisite: Undergraduate level knowledge of linear algebra, statistics, and a programming language.
Course Learning Outcomes

- 1. Be able to analyze and predict the behavior of image formation, transformation, and recognition algorithms
- 2. Be able to design, develop, and evaluate algorithms for specific applications
- 3. Be able to use software tools to implement computer vision algorithms
- 4. Communicate (in oral and written form) methods and results to a technical audience

CSCI508. ADVANCED TOPICS IN PERCEPTION AND COMPUTER VISION. 3.0 Semester Hrs.
Equivalent with EENG508,
(II) This course covers advanced topics in perception and computer vision, emphasizing research advances in the field. The course focuses on structure and motion estimation, general object detection and recognition, and tracking. Projects will be emphasized, using popular software tools. Prerequisites: EENG507 or CSCI507. 3 hours lecture; 3 semester hours.
Course Learning Outcomes

- 1. Be able to review the literature on computer vision and create a critical review
- 2. Be able to design, develop, and evaluate algorithms for specific applications
- 3. Be able to use software tools to implement computer vision algorithms
- 4. Communicate (in oral and written form) methods and results to a technical audience

CSCI520. SOFTWARE ENGINEERING SYSTEMS DEVELOPMENT I. 3.0 Semester Hrs.
This is the first course of the two-semester sequence. In this course you will be addressing realistic, open-ended design changes for clients in interdisciplinary (when appropriate) teams. The design projects are chosen by the course faculty to develop creativity, encourage use of appropriate design methods and application of technical skills. You are expected to successfully address the given challenge. The devised solution in partnership with the client must be within the constraints of elements such as cost, sustainability, manufacturability/deployment. Projects many be both hardware and software focused. By the end of the first semester, teams should have a fully defined the system architecture, specifications, and operation. Solution components have been modeled, analyzed and selected and a proof-of-concept protype developed to demonstrate the solution to the client.
Course Learning Outcomes

- Develop and maintain a project management plan to address a software engineering design challenge that includes a clear definition of done and detailed project schedule.
- Formulate a clear, concise set of correctly constructed engineering metrics/constraints and specifications for a software/hardware engineering design challenge.
- Develop and graphically communicate multiple innovative high-level design solutions to a given software/hardware engineering challenge.
- Utilize decision methods to select the most promising solutions(s) that meet(s) constraints and requirements for the given design challenge.
- Develop a proof-of-concept protype(s) for the critical and high-risk functional elements of the design challenge.
- Develop a project completion plan and execute Agile project development methodology that allows the team to successfully meet the project goals.
- Apply appropriate technical knowledge to solve a design challenge.
- Communicate your engineering design, research, analysis, decision making and solution via professional, technical, and industry appropriate written documentation packages.
- Communicate technical information verbally as part of a sprint or design review and synthesize feedback from stakeholders and clients.

- Identify and analyze the ethical, environmental, societal and/or economic impact of considered design solutions.
- Function effectively on a team whose members together provide leadership, create a collaborative and inclusive environment, establish goals, plan tasks and meet design challenge objectives.
- Evaluate your success as a designer by reflecting on your experience and proposing strategies for future self-improvement.

CSCI521. SOFTWARE ENGINEERING SYSTEMS DEVELOPMENT II. 3.0 Semester Hrs.

This is the second course of the two-semester sequence. In this course you will be addressing realistic, open-ended design changes for clients in interdisciplinary (when appropriate) teams. The design projects are chosen by the course faculty to develop creativity, encourage use of appropriate design methods and application of technical skills. You are expected to successfully address the given challenge. The devised solution in partnership with the client must be within the constraints of elements such as cost, sustainability, manufacturability/deployment. Projects many be both hardware and software focused. By the end of the first semester, teams should have a fully defined the system architecture, specifications, and operation. Solution components have been modeled, analyzed and selected and a proof-of-concept prototype developed to demonstrate the solution to the client.

CSCI532. ROBOT ETHICS. 3.0 Semester Hrs.

(II) This course explores ethical issues arising in robotics and human-robot interaction through philosophical analysis, scientific experimentation, and algorithm design. Topics include case studies in lethal autonomous weapon systems, autonomous cars, and social robots, as well as higher-level concerns including economics, law, policy, and discrimination. Graduate enrollees will additionally participate in and report on the results of empirical and computational robot ethics research, with the goal of developing publishable works. Prerequisite: Graduate student standing.

Course Learning Outcomes

- 1 - Understand the basic ethical theories, concepts, tools, and frameworks for analyzing the social and ethical ramifications of robotics
- 2 - Be able to critically examine the ethical significance of the use of robotics in daily and technical fields including human-robot interaction, medicine, relationship, military, etc.
- 3 - Develop a critical attitude toward the role of robotics in shaping human society including human perceptions and behaviors
- 4 - Be able to use the theories, concepts, tools, and frameworks learned from this class to critically examine emerging robot ethics issues in the society.
- 5 - Understand the tradeoffs underlying the design of autonomous moral agents.
- 6 - Conduct robot ethics research grounded in both human-subject experimentation and algorithm development.

CSCI534. ROBOT PLANNING AND MANIPULATION. 3.0 Semester Hrs.

An introduction to planning in the context of robotics covering symbolic and motion planning approaches. Symbolic computation, symbolic domains, and efficient algorithms for symbolic planning; Robot kinematics, configuration spaces, and algorithms for motion planning. Applications of planning will focus on manipulation problems using robot arms.

Course Learning Outcomes

- 1 - Implement algorithms for symbolic computation
- 2 - Construct symbolic planning domains for new scenarios
- 3 - Implement algorithms for symbolic planning via constraint-solving and heuristic search
- 4 - Implement algorithms for sampling-based motion planning
- 5 - Construct kinematic models of robot manipulators
- 6 - Analyze planning algorithms for key properties: correctness, completeness, optimality
- 7 - Evaluate the suitability of different planning approaches and apply appropriate algorithms to new planning scenarios
- 8 - Communicate implementations, analysis, and evaluation in written and oral form

CSCI536. HUMAN-ROBOT INTERACTION. 3.0 Semester Hrs.

Human-Robot Interaction is an interdisciplinary field at the intersection of Computer Science, Robotics, Psychology, and Human Factors, that seeks to answer a broad set of questions about robots designed to interact with humans (e.g., assistive robots, educational robots, and service robots), such as: (1) How does human interaction with robots differ from interaction with other people? (2) How does the appearance and behavior of a robot change how humans perceive, trust, and interact with that robot? And (3) How can we design and program robots that are natural, trustworthy, and effective? Accordingly, In this course, students will learn (1) how to design interactive robots, (2) the algorithmic foundations of interactive robots; and (3) how to evaluate interactive robots. To achieve these learning objectives, students will read and present key papers from the HRI literature, complete an individual final project tailored to their unique interests and skillsets, and complete a group project in which they will design, pilot, and evaluate novel HRI experiments, with in-class time expected to be split between lecture by the instructor, presentations by students, and either collaborative active learning activities or discussions with researchers in the field. Prerequisite: Data Structures, Probability and Statistics or equivalent.

Course Learning Outcomes

- Understand the theoretical foundations and critical application domains of human-robot interaction.
- Employ design techniques to design interactive robots.
- Design human-subject experiments to evaluate interactive robots.
- Perform qualitative and quantitative analysis on the results of human-robot interaction experiments.

CSCI538. AUGMENTED REALITY. 3.0 Semester Hrs.

Augmented reality is the process of augmenting the real world with virtual content, typically with hand-held or head-mounted displays. This course will provide an overview of augmented reality, including applications, technical foundations (computer vision, optics, graphics, and human factors), and recent developments in the field. Projects will be emphasized, using popular software tools. Augmented reality is a systems-thinking problem where it is helpful to have a good understanding of several areas, including computer vision, optics, graphics, and human-computer interaction. Projects will provide students with opportunities to gain a deeper understanding of these areas, as well as to build a portfolio for possible future careers in augmented reality.

Course Learning Outcomes

- At the completion of the course, you will be able to:

CSCI542. SIMULATION. 3.0 Semester Hrs.

(I) Advanced study of computational and mathematical techniques for modeling, simulating, and analyzing the performance of various systems. Simulation permits the evaluation of performance prior to the implementation of a system; it permits the comparison of various operational alternatives without perturbing the real system. Topics to be covered include simulation techniques, random number generation, Monte Carlo simulations, discrete and continuous stochastic models, and point/interval estimation. Offered every other year. 3 hours lecture; 3 semester hours.

CSCI544. ADVANCED COMPUTER GRAPHICS. 3.0 Semester Hrs.
Equivalent with MATH544,
This is an advanced computer graphics course in which students will learn a
variety of mathematical and algorithmic techniques that can be used to solve
fundamental problems in computer graphics. Topics include global illumination,
GPU programming, geometry acquisition and processing, point based graphics
and non-photorealistic rendering. Students will learn about modern rendering and
geometric modeling techniques by reading and discussing research papers and
implementing one or more of the algorithms described in the literature.

CSCI555. GAME THEORY AND NETWORKS. 3.0 Semester Hrs.
Equivalent with CSCI455,
(II) An introduction to fundamental concepts of game theory with a focus on the applications in networks. Game theory is the study that analyzes the strategic interactions among autonomous decision-makers. Originated from economics. Influenced many areas in Computer Science, including artificial intelligence, e-commerce, theory, and security and privacy. Provides tools and knowledge for modeling and analyzing real-world problems. Prerequisites: CSCI406 Algorithms. 3 hours lecture; 3 semester hours.
Course Learning Outcomes

- 1. Understand the basic concepts in game theory
- 2. Be able to model and analyze real-world problems as games
- 3. Learn about the game theoretic applications in networks
- 4. Have the opportunity to apply game theory to other fields

CSCI560. FUNDAMENTALS OF COMPUTER NETWORKS. 3.0 Semester Hrs.
This fully online course provides an introduction to fundamental concepts in the design and implementation of computer communication networks, their protocols, and applications. Topics include overview of network architectures, applications, network programming interfaces (e.g., sockets), transport, congestion, routing, and data link protocols, addressing, local area networks, wireless networks, and network security. Examples are drawn primarily from the Internet (e.g., TCP, UDP, and IP) protocol suite. If you have taken CSCI 471, you will not receive credit for this course. Prerequisite: CSCI442.
Course Learning Outcomes

- 1) Explain how packet switching works
- 2) Compute and identify potential packet delays in a packet-switched network when provided specific parameters of network conditions.
- 3) Articulate in laymen's terms the following concepts or processes: a) essential principles of a transport layer protocol (reliable data transfer, flow control, congestion control), b) data link layer services and multiple access techniques, c) how mobility is managed in cellular networks, and d) network security issues and provide examples of a few methods that address them
- 4) Apply routing algorithms to find shortest paths for network-layer packet delivery
- 4) Apply routing algorithms to find shortest paths for network-layer packet delivery
- 5) Apply Wireshark, a network sniffing tool, to observe and analyze behaviors of the networking protocols studied in this course
- 6) Design and implement distributed applications using the socket APIs and support for TCP/UDP communications between end hosts

CSCI561. THEORY OF COMPUTATION. 3.0 Semester Hrs.
(I) An introduction to abstract models of computation and computability theory; including finite automata (finite state machines), pushdown automata, and Turing machines. Language models, including formal languages, regular expressions, and grammars. Decidability and undecidability of computational problems. 3 hours lecture; 3 semester hours. Prerequisite: CSCI406.
Course Learning Outcomes

- Design and implement various types of automata, including finite automata, pushdown automata, and Turing machines.
- Model and solve computational problems using formal language representations and algorithms.
- Communicate effectively using mathematically precise terminology, particularly about topics in language and complexity theory.
- Formally prove capabilities and limits, such as decidability and complexity of new problems, based on computational classes.
- Evaluate potentially effective (or ineffective) approaches for new problems based on language and complexity class.
- Apply reduction techniques to show the equivalence between different computational problems.

CSCI562. APPLIED ALGORITHMS AND DATA STRUCTURES. 3.0 Semester Hrs.
(II) Industry competitiveness in certain areas is often based on the use of better algorithms and data structures. The objective of this class is to survey some interesting application areas and to understand the core algorithms and data structures that support these applications. Application areas could change with each offering of the class, but would include some of the following: VLSI design automation, computational biology, mobile computing, computer security, data compression, web search engines, geographical information systems. 3 hours lecture; 3 semester hours. Prerequisite: CSCI406.
Course Learning Outcomes

- Communicate effectively on matters related to algorithms and data structures to a range of audiences using a range of media.
- Solve moderately open-ended problems related to graphs, NP completeness, approximation algorithms, and geometric algorithms.
- Demonstrate attention to detail (with respect to definitions, terminology, notation, language) as measured by performance on multiple-choice and true-false questions on online-quizzes.
- Execute a sequence of increasingly difficult (but guided) research related tasks related to an important application area that requires.
- Function effectively as a team member with effectiveness being determined by peer ratings, self-assessment and instructor observations.

CSCI563. PARALLEL COMPUTING FOR SCIENTISTS AND ENGINEERS. 3.0 Semester Hrs.
(I) Students are taught how to use parallel computing to solve complex scientific problems. They learn how to develop parallel programs, how to analyze their performance, and how to optimize program performance. The course covers the classification of parallel computers, shared memory versus distributed memory machines, software issues, and hardware issues in parallel computing. Students write programs for state of the art high performance supercomputers, which are accessed over the network. Prerequisite: Programming experience in C. 3 hours lecture; 3 semester hours.
Course Learning Outcomes

- Develop parallel programs to solve complex scientific problems using both shared memory and distributed memory models.
- Analyze and optimize the performance of parallel algorithms to achieve efficient computational solutions.
- Classify and differentiate between various parallel computing architectures and understand the associated hardware and software challenges.
- Gain hands-on experience with high-performance supercomputers by writing, testing, and debugging parallel applications over networked environments.

CSCI564. ADVANCED COMPUTER ARCHITECTURE. 3.0 Semester Hrs.
The objective of this class is to gain a detailed understanding about the options available to a computer architect when designing a computer system along with quantitative justifications for the options. All aspects of modern computer architectures including instruction sets, processor design, memory system design, storage system design, multiprocessors, and software approaches will be discussed. Prerequisite: CSCI341. 3 hours lecture; 3 semester hours.
Course Learning Outcomes

- Understand quantitative methods to evaluate system performance, including metrics and benchmarks.
- Analyze the design and optimization of memory hierarchies, including caches and virtual memory.
- Understand CPU pipelines, focusing on throughput and instruction scheduling.
- Understand the techniques for speculative execution and branch prediction.
- Understand the GPU architecture.

CSCI565. DISTRIBUTED SYSTEMS. 3.0 Semester Hrs.
This course discusses concepts, techniques, and issues in developing distributed systems in large scale networked environment. Topics include theory and systems level issues in the design and implementation of distributed systems.
Course Learning Outcomes

- Analyze complexity of distributed algorithms.
- Prove correctness of distributed algorithms.
- Implement virtual clocks.
- Implement and evaluate distributed minimum spanning tree algorithms.
- Implement distributed mutual exclusion algorithms.
- Compare and contrast distributed leader election algorithms.
- Implement different data-centric and client-centric consistency models.
- Review recent distributed systems research.

CSCI568. DATA MINING. 3.0 Semester Hrs.
This course is an introductory course in data mining. It covers fundamentals of data mining theories and techniques. We will discuss association rule mining and its applications, overview of classification and clustering, data preprocessing, and several application specific data mining tasks. We will also discuss practical data mining using a data mining software. Project assignments include implementation of existing data mining algorithms, data mining with or without data mining software, and study of data mining related research issues. 3 hours lecture; 3 semester hours.

CSCI571. ARTIFICIAL INTELLIGENCE. 3.0 Semester Hrs.
Artificial Intelligence (AI) is the subfield of computer science that studies how to automate tasks for which people currently exhibit superior performance over computers. Historically, AI has studied problems such as machine learning, language understanding, game playing, planning, robotics, and machine vision. AI techniques include those for uncertainty management, automated theorem proving, heuristic search, neural networks, and simulation of expert performance in specialized domains like medical diagnosis. This course provides an overview of the field of Artificial Intelligence. Particular attention will be paid to learning the LISP language for AI programming. Prerequisite: CSCI262.

CSCI572. COMPUTER NETWORKS II. 3.0 Semester Hrs.
This course explores how computer networking is evolving to support new environments, and challenges in building networked systems that are simultaneously highly robust, efficient, flexible, and secure. Detailed topics include wireless and mobile networks, multimedia networking, and network security. In addition, recent research and developments are also studied, which include mobile sensing, Internet of Things (IoT), social computing and networks, mobile ad-hoc networks, wireless sensor networks, software defined networking, and future Internet architecture.
Course Learning Outcomes

- Articulate wireless networking challenges.
- Describe wireless MAC layer protocols.
- Describe the design of WiFi networks and Cellular networks.
- Use network sniffing tools to characterize the impact of physical environments on wireless signal propagation.
- Use a network simulator to evaluate the performance of routing protocols for wireless ad-hoc networks.
- Analyze WiFi's MAC layer protocol in different scenarios using a network simulator.
- Review recent wireless networking research.

CSCI573. ROBOT PROGRAMMING AND PERCEPTION. 3.0 Semester Hrs.
Equivalent with CSCI473,
In this class students will learn the basics of integrated robot system programming and the design and use of algorithms for robot perception. Students will learn how to use the ROS robot middleware for the design of robot systems for perceiving and navigating the world; develop reinforcement learning based models for perception-informed autonomous navigation; and develop computational models for 3D robot perception and perceptual representation of human data.
Course Learning Outcomes

- 1. Explain the basic concepts in human-centered robotics
- 2. Model and analyze human behaviors for human-robot interaction applications
- 3. Recognize the cutting-edge human-centered robotics reasearch and applications
- 4. Apply the learned knowledge to other fields

CSCI574. THEORY OF CRYPTOGRAPHY. 3.0 Semester Hrs.
Equivalent with MATH574,
(I) Students will draw upon current research results to design, implement and analyze their own computer security or other related cryptography projects. The requisite mathematical background, including relevant aspects of number theory and mathematical statistics, will be covered in lecture. Students will be expected to review current literature from prominent researchers in cryptography and to present their findings to the class. Particular focus will be given to the application of various techniques to real-life situations. The course will also cover the following aspects of cryptography: symmetric and asymmetric encryption, computational number theory, quantum encryption, RSA and discrete log systems, SHA, steganography, chaotic and pseudo-random sequences, message authentication, digital signatures, key distribution and key management, and block ciphers. Prerequisite: CSCI262. 3 hours lecture, 3 semester hours.
Course Learning Outcomes

- Explain the basic terms and concepts in cryptography.
- Correlate number theory with cryptography.
- Analyze the limitations of classical cryptographic algorithms.
- Compare the similarities and differences between symmetric and asymmetric encryption algorithms.
- Assess the security strength of cryptographic techniques.
- Choose appropriate cryptographic techniques such as hashing, message authentication, and digital signature for addressing real-world security and privacy problems.
- Critique research work in applied cryptography.

CSCI575. ADVANCED MACHINE LEARNING. 3.0 Semester Hrs.
Machine learning is the study of computer algorithms that improve automatically through experience. Machine learning systems do not have to be programmed by humans to solve a problem; instead, they essentially program themselves based on examples of how they should behave, or based on trial and error experience trying to solve the problem. This course aims at provide students with an understanding of the capabilities of machine learning (especially for deep learning due to its state-of-the-art performance for predicting and understanding data), and the knowledge to formulate the real-world problem to solve it effectively by a combination of computational idea motivations, learning theories, mathematical and optimization backgrounds/tools. Prerequisite: CSCI470.

CSCI576. DEEP LEARNING. 3.0 Semester Hrs.
Machine Learning is a key component of Artificial Intelligence that allows computers to generate accurate outcomes without being explicitly programmed. Over the past decade, machine learning has powered speech recognition, augmented reality, self-driving vehicles, and biometric authentications. The purpose of this course is to bridge general machine learning concepts to real world applications. This course provides a broad introduction to machine learning (supervised learning, unsupervised learning, feature learning), deep learning, computer vision, natural language processing, and reinforcement learning. During the course, students will design a project that applies concepts from class to several diGerent real-world problems and challenges and create applications to solve them. Prerequisite: CSCI470.

CSCI577. ADVANCED ELEMENTS OF GAMES AND GAME DEVELOPMENT. 3.0 Semester Hrs.
This course provides an overview of computer and video game development along with practical game projects designed to introduce the student to the computer entertainment industry. Topics will include the nature of games, the game player, game play, game design, game mechanics, story and character, game worlds, interface and the game development process. Students will be required to develop code in existing game engines (GameMaker, Unity, and Unreal). Students will also work on a game engine from scratch, using a hybrid software design pattern that is an object-oriented hierarchy with components. The majority of the course will be dedicated to a semester-long game project where students will create a complete game with a teammate using Agile principles, a backlog, a design document, demos, and a final product with commercial video. Prerequisite: CSCI 306.
Course Learning Outcomes

CSCI578. BIOINFORMATICS. 3.0 Semester Hrs.
This course contains 4 main modules organized around different data types produced by biotechnological assays, i.e., DNA/protein sequences, protein structures, gene activity levels, relationships between biological entities. Students will have a chance to interact with different public datasets and experiment with different tools and platforms with hands-on exercises. The main objective of the course is to provide the student with a solid foundation for conducting further research in bioinformatics. By the end of the course, the students will have learned: the bioinformatics terminology, main bioinformatics problems, and the key computational methods and tools used in bioinformatics. Prerequisites: Prior experience with introductory programming concepts.
Course Learning Outcomes

- Construct a systems level abstraction of the dynamics of a cell by relating components of the cell such as genes, proteins, and metabolic reactions
- Design and implement algorithms for solving pairwise, multiple sequence alignments, and search sequence databases using sequences from NCBI GenBank or Uniprot
- Design and implement algorithms for constructing phylogenetic trees for a set of biological sequences and apply them on real datasets on sequence families, such as SARS-CoV-2 variants
- Apply existing tools to predict protein structures from sequence and compare multiple protein structures using sequences from Uniprot and structures from the Protein Data Bank
- Interpret and analyze transcriptome data such as microarray data or RNA-Seq data by to reveal clusters of genes/samples and identify differentially expressed genes
- Design and implement algorithms for analyzing biological networks such as protein-protein interaction networks, gene-regulatory networks, and metabolic networks

CSCI580. ADVANCED HIGH PERFORMANCE COMPUTING. 3.0 Semester Hrs.
This course provides students with knowledge of the fundamental concepts of high performance computing as well as hands-on experience with the core technology in the field. The objective of this class is to understand how to achieve high performance on a wide range of computational platforms. Topics will include sequential computers including memory hierarchies, shared memory computers and multicore, distributed memory computers, graphical processing units (GPUs), cloud

and grid computing, threads, OpenMP, message passing (MPI), CUDA (for GPUs), parallel file systems, and scientific applications. 3 hours lecture; 3 semester hours.

Course Learning Outcomes

- Develop skills to design, implement, and debug parallel programs using frameworks like MPI and OpenMP.
- Learn to analyze and optimize computational performance on multi-core and distributed systems.
- Understand and apply algorithms that efficiently scale across high-performance computing architectures.
- Gain familiarity with HPC tools, libraries, and systems to solve complex computational problems effectively.

CSCI581. QUANTUM PROGRAMMING. 3.0 Semester Hrs.

This course serves as an introduction to programming quantum computers. Students will receive an in depth education in quantum algorithms and their design, and then break into teams to learn the API of a commercially available quantum computing system. They will use this system to write and test simple quantum algorithms, and debug their code to improve its performance against noise and other error sources. Prerequisite: PHGN519.

Course Learning Outcomes

- Building on the education received in Fundamentals of Quantum Information, students will learn:
- 1. How to write and implement simple quantum algorithms,
- 2. Understand mechanisms by which a quantum speedup can be obtained
- 3. Use and execute code on an API of publicly available quantum hardware (e.g. IBM's qiskit, Rigetti's Forest, Google's forthcoming Cirq, and many more)
- 4. “Debug” their code to improve its performance given the realities of noise and gate error

CSCI582. COMPUTING BEYOND CPU'S. 3.0 Semester Hrs.

CPUs are no longer fast enough to execute demanding tasks in emerging domains such as artificial intelligence, robotics and mobile computing. Most modern systems come with specialized, i.e., “beyond the CPU”, processing units such as GPUs, neural network processors, and other domain-specific accelerators. This class will provide students with knowledge and hands-on experience with accelerators. We will dig deep into the world of specialized computing and: (a) teach students the fundamental computer architecture concepts that lead to the development of accelerators, (b) let the students know about modern accelerators, their internal architectures, and how to program them to solve many prevalent computational problems in an energy and performance efficient way, and (c) provide students a hands-on experience by implementing and running benchmarks on an accelerator of their choice and perform a detailed performance analysis. The course assignments will include weekly paper reading, mini programming assignments, paper presentations (one or two times in the semester) and a semester project that will be carried out on an accelerator of student's choice. The students will be given access to a diverse range of modern heterogeneous processing units that are designed to accelerate mobile, cloud and autonomous workloads. Prerequisite: CSCI 341.

Course Learning Outcomes

- Explore the principal design concepts of domain-specific accelerators, contrast them to traditional CPU architectures, understand how they are programmed, and comprehend their role and importance in modern computing.
- Improve critical thinking and analysis abilities on accelerated computing research and practice technical presentation of concepts and papers in both spoken and written forms.
- Measure and analyze performance of the state-of-the-art system-on-chips for autonomous and mobile computing using industry-standard parallel-programming languages, paradigms, and APIs.

CSCI583. IOT SECURITY AND PRIVACY. 3.0 Semester Hrs.

The use of IoT (Internet-of-Things) devices in U.S. homes is becoming increasingly popular. These smart devices, however, come with a range of security and privacy risks that need to be addressed. This course focuses on teaching the basics and advanced concepts of IoT security and privacy from a data-driven perspective. It covers topics such as networking and Internet security, as well as security research based on data analysis. Students will be required to read and present academic papers from various disciplines such as computer science, psychology, policy, and law. The course also includes hands-on projects that allow students to independently investigate real-world security and privacy issues of IoT devices and suggest practical solutions to fix them. Through this course, students will develop a comprehensive understanding of the security and privacy challenges posed by IoT devices and learn how to approach these issues in a data-driven and evidence-based manner.

Course Learning Outcomes

- Develop a comprehensive understanding of the security and privacy challenges posed by IoT devices.
- Approach these issues in a data-driven and evidence-based manner.
- Read, understand, present, and critique IoT academic papers.
- Analyze, formulate and address IoT security and privacy research problems.
- Master cutting-edge IoT data analysis techniques using AI models (e.g., generative and embedded AI).

CSCI585. INFORMATION SECURITY PRIVACY. 3.0 Semester Hrs.

This course provides an introduction to the principles and best practices in information security and privacy. Lectures will include basic concepts of information security and privacy, fundamental security design principles, major topics in security and privacy, essential knowledge and skills, risk assessment and mitigation, policy development, and so on. In the classroom, students will also present and discuss a list of recent or classic research papers corresponding to the major topics in security and privacy. Outside of the classroom, students will work on homework assignments, security lab exercises, quizzes, research paper summaries, and a course project. Prerequisite: (CSCI262 or CSCI220) and CSCI341.

Course Learning Outcomes

- Explain the fundamental security and privacy protection principles, techniques, and practices.
- Describe risk mitigation, legal requirements, and ethical considerations in cybersecurity.
- Analyze real-world security and privacy problems in modern computing environments.
- Practice security and privacy protection tools and techniques.
- Critique existing security and privacy protection techniques or studies.

CSCI587. CYBER PHYSICAL SYSTEMS SECURITY. 3.0 Semester Hrs.
(II) This course aims to build a solid foundation for students to identify, analyze, and evaluate real-world security and privacy problems in Cyber Physical Systems, as well as to design and develop secure and usable solutions for addressing these problems. It focuses on the important security and privacy research topics in representative Cyber Physical Systems such as wireless sensor networks, smart grids, autonomous automotive systems, and robotic systems. It also includes the discussion of the protection of the nation's critical infrastructures such as Food, Health, Water, Energy, Finance, Communication, Manufacturing, Government, and Transportation. The format of the course includes introductory discussions, research paper reading, summaries, and discussions, as well as research projects. 3 hours lecture; 3 semester hours.
Course Learning Outcomes

- 1. Students will be able to explain the essential security and privacy protection requirements in representative Cyber Physical Systems.
- 2. Students will be able to analyze security and privacy problems in representative Cyber Physical Systems.
- 3. Students will be able to analyze and evaluate research articles in Cyber Physical Systems security and privacy protection.
- 4. Students will be able to design usable solutions for addressing security and privacy problems in representative Cyber Physical Systems.

CSCI597. SUMMER PROGRAMS. 0-6 Semester Hr.

CSCI598. SPECIAL TOPICS. 0-6 Semester Hr.
(I, II, S) Pilot course or special topics course. Topics chosen from special interests of instructor(s) and student(s). Usually the course is offered only once, but no more than twice for the same course content. Prerequisite: none. Variable credit: 0 to 6 credit hours. Repeatable for credit under different titles.

CSCI598. SPECIAL TOPICS. 0-6 Semester Hr.

CSCI598. SPECIAL TOPICS. 0-6 Semester Hr.

CSCI598. SPECIAL TOPICS. 0-6 Semester Hr.

CSCI598. SPECIAL TOPICS. 0-6 Semester Hr.

CSCI598. SPECIAL TOPICS. 1-6 Semester Hr.

CSCI598. SPECIAL TOPICS. 1-6 Semester Hr.

CSCI598. SPECIAL TOPICS. 1-6 Semester Hr.

CSCI598. SPECIAL TOPICS. 0-6 Semester Hr.

CSCI598. SPECIAL TOPICS. 0.5-6 Semester Hr.

CSCI598. SPECIAL TOPICS. 0.5-6 Semester Hr.

CSCI598. SPECIAL TOPICS. 0.5-6 Semester Hr.

CSCI599. INDEPENDENT STUDY. 0.5-6 Semester Hr.
(I, II, S) Individual research or special problem projects supervised by a faculty member, also, when a student and instructor agree on a subject matter, content, and credit hours. Prerequisite: “Independent Study” form must be completed and submitted to the Registrar. Variable credit: 0.5 to 6 credit hours. Repeatable for credit under different topics/experience and maximums vary by department. Contact the Department for credit limits toward the degree.

CSCI599. INDEPENDENT STUDY. 0.5-6 Semester Hr.

CSCI599. INDEPENDENT STUDY. 0.5-6 Semester Hr.

CSCI599. INDEPENDENT STUDY. 0.5-6 Semester Hr.

CSCI599. INDEPENDENT STUDY. 0.5-6 Semester Hr.

CSCI599. INDEPENDENT STUDY. 0.5-6 Semester Hr.

CSCI599. INDEPENDENT STUDY. 0.5-6 Semester Hr.

CSCI599. INDEPENDENT STUDY. 0.5-6 Semester Hr.

CSCI599. INDEPENDENT STUDY. 0.5-6 Semester Hr.

CSCI599. INDEPENDENT STUDY. 0.5-6 Semester Hr.

CSCI599. INDEPENDENT STUDY. 0.5-6 Semester Hr.

CSCI599. INDEPENDENT STUDY. 0.5-6 Semester Hr.

CSCI599. INDEPENDENT STUDY. 0.5-6 Semester Hr.

CSCI599. INDEPENDENT STUDY. 0.5-6 Semester Hr.

CSCI599. INDEPENDENT STUDY. 0.5-6 Semester Hr.

CSCI599. INDEPENDENT STUDY. 0.5-6 Semester Hr.

CSCI691. GRADUATE SEMINAR. 1.0 Semester Hr.
Presentation of latest research results by guest lecturers, staff, and advanced students. Prerequisite: none. 1 hour seminar; 1 semester hour. Repeatable for credit to a maximum of 12 hours.

CSCI698. SPECIAL TOPICS. 0-6 Semester Hr.
(I, II, S) Pilot course or special topics course. Topics chosen from special interests of instructor(s) and student(s). Usually the course is offered only once, but no more than twice for the same course content. Prerequisite: none. Variable credit: 0 to 6 credit hours. Repeatable for credit under different titles.

CSCI699. INDEPENDENT STUDY. 0.5-6 Semester Hr.
(I, II, S) Individual research or special problem projects supervised by a faculty member, also, when a student and instructor agree on a subject matter, content, and credit hours. Prerequisite: “Independent Study” form must be completed and submitted to the Registrar. Variable credit: 0.5 to 6 credit hours. Repeatable for credit under different topics/experience and maximums vary by department. Contact the Department for credit limits toward the degree.

CSCI700. MASTERS PROJECT CREDITS. 1-6 Semester Hr.
(I, II, S) Project credit hours required for completion of the non-thesis Master of Science degree in Computer Science (Project Option). Project under the direct supervision of a faculty advisor. Credit is not transferable to any 400, 500, or 600 level courses. Repeatable for credit.

CSCI707. GRADUATE THESIS / DISSERTATION RESEARCH CREDIT. 1-15 Semester Hr.
(I, II, S) GRADUATE THESIS/DISSERTATION RESEARCH CREDIT Research credit hours required for completion of a Masters-level thesis or Doctoral dissertation. Research must be carried out under the direct supervision of the student's faculty advisor. Variable class and semester hours. Repeatable for credit.

Professor and Department Head

Iris Bahar

Professors

Qi Han

Dinesh Mehta

Associate Professors

Hua Wang

Bo Wu

Thomas Williams

Dejun Yang

Chuan Yue

Assistant Professors

Mehmet Belviranli

Dong Chen

Neil Dantam

Gabriel Fierro

C. Estelle Smith

Teaching Professors

Tolga Can

Vibhuti Dave

Jeffrey Paone

Christopher Painter-Wakefield

Teaching Associate Professors

Wendy Fisher, Assistant Department Head

Kathleen Kelly

Teaching Assistant Professors

Amelia Read

Robert Thompson

Zibo Wang

Professors of Practice

Mark Baldwin

Phil Romig

Visiting Teaching Associate Professor

Keith Hellman

Emeriti Professors

Tracy Camp , Emeritus Professor

William Hoff, Emeritus Associate Professor

Cyndi Rader, Emeritus Teaching Professor