Master of Science with a major in Computer Science
Course of Study

General Requirements

Thesis Option

Core

Requirements | Credit Hours
--- | ---
CSCI 5100 | Algorithm Analysis 3
CSCI 5110 | Theory of Computation 3
CSCI 5120 | Advanced Topics in Computer Security 3
CSCI 5130 | Software Engineering 3
CSCI 6900 | Introduction to Research 3
CSCI 6910 | Master’s Thesis Research 9

**Core Total Hours** 24

Electives

Credit Hours 6

**Total Credit Hours** 30

Non-Thesis Option

Core

Requirements | Credit Hours
--- | ---
CSCI 5100 | Algorithm Analysis 3
CSCI 5110 | Theory of Computation 3
CSCI 5120 | Advanced Topics in Computer Security 3
CSCI 5130 | Software Engineering 3
CSCI 6900 | Introduction to Research 3

**Core Total Hours** 15

Electives

Credit Hours 15

**Total Credit Hours** 30

Electives

- CSCI 5300: Programming Languages
- CSCI 5310: Proof Theory
- CSCI 5320: Verification of Software
- CSCI 5330: Artificial Intelligence
- CSCI 5340: Machine Learning
- CSCI 5350: Network & Distributed Systems
- CSCI 6100: Cyber-physical System
- CSCI 6950: Selected Topics
Course Descriptions

CSCI 5100: Algorithm Analysis, 3 Credits
This course teaches abstract techniques for analyzing and designing algorithms. Tools include cost models, run-time analysis, dynamic programming, and the study of upper and lower bounds. The goals are to understand and assess classical algorithms on various data structure (including graphs), as well as to sharpen problem solving skills and development of new algorithms.

CSCI 5110: Theory of Computation, 3 Credits
In this course, we will seek to understand what is and is not possible with modern day computers by using mathematical models to fine-grain study various forms of computation. We will study languages, Turing machines, undecidability, the time complexity classes such as P, NP, NP-complete, space complexity classes such as L and NL, the Cook-Levin theorem, reductions, polynomial hierarchy, randomized algorithms and randomized complexity classes such as BPP, approximation algorithms and hardness of approximation.

CSCI 5120: Advanced Topics in Computer Security, 3 Credits
This course examines the foundational topics in computer security, such as cryptography, human computer interaction, virtualization, cloud computing, and network protocols. There is a lab component to the course.

CSCI 5130: Software Engineering, 3 Credits
This course covers concepts in software engineering. The use of software engineering methodologies and tools as mechanisms for enhancing software quality attributes. The classification, evaluation and selection of software engineering techniques.

CSCI 5300: Programming Languages, 3 Credits
Programming languages are a fundamental aspect of computer science, and as computer science and society evolve the need for more advanced programming languages rises. This course will provide the understanding of how to formally design and analyze programming languages using formal models. The student will gain an understanding of the syntax and semantics of programming languages to gain a deeper understanding of correctness of programs, software verification, and the wider notion of formal methods. Furthermore, through course projects the student will gain an understanding of programming in various paradigms, and in the implementation of programming languages.

CSCI 5310: Proof Theory, 3 Credits
This course teaches the theory of proofs studied as a formal and mathematical object. The language of proofs emerged as a natural place where programming languages, category theory, and complexity meet, and is becoming increasingly popular in the study of type systems, denotational semantics, concurrency theory, implicit complexity, and higher-order model checking to name a few. Tools include mathematical abstraction, theorem provers, the representation of proofs in multiple formalisms (sequent calculus, natural deduction, proof nets and their semantics) and multiple systems, as well as studying their relative expressivity and complexity. The goals are to excel in formal reasoning, have a precise and actionable understanding of the expressivity of a
system compared to others, and improve mathematical reasoning and the capacity to navigate between multiple representations of the same proof.

**CSCI 5320: Verification of Software, 3 Credits**
The course covers formal methods theory and practice, in particular as applied to the specification and verification of software.

**CSCI 5330: Artificial Intelligence, 3 Credits**
Presentation of artificial intelligence as a coherent body of ideas and methods to acquaint the student with the basic programs in the field and their underlying theory. Concepts and methods include topics such as informed, adversarial, and probabilistic search, knowledge representation and symbolic reasoning, knowledge engineering and building problem solvers.

**CSCI 5340: Machine Learning, 3 Credits**
This is a survey of machine learning theory and practice. The course covers the fundamental concepts and algorithms that allow computers to improve performance over time. Techniques enabling supervised learning, unsupervised learning and reinforcement learning will be covered.

**CSCI 5350: Network & Distributed Systems, 3 Credits**
This is an introduction to algorithmic aspects of distributed computing. The topics considered are those that arise in systems comprised of loosely coupled, heterogeneous and failure-prone processing units. The range of applications starts at wide-area networks, goes through clusters of workstations connected by local-area networks, to multi-processor shared-memory machines. The relevant properties of solutions reflect the communication mechanisms (message passing, shared memory), the algorithmic constraints (deterministic, randomized), the timing models (synchronous, asynchronous), and the types of failures (crashes, Byzantine). The algorithmic goals to achieve include sharing resources in a fair manner, providing fault-tolerance, and maintaining global consistency of computations. The specific problems include symmetry breaking, consensus, resource allocation, renaming, and synchronization.

**CSCI 6100: Cyber-physical System, 3 Credits**
This course strives to identify and introduce the durable intellectual ideas of embedded systems as a technology and as a subject of study. The emphasis is on modeling, design, and analysis of cyber-physical systems, which integrate computing, networking, and physical processes.

**CSCI 6900: Introduction to Research, 3 Credits**
In this course, students are introduced to methods for conducting scientific research within computer science. Emphasis is placed on research question formation, literature review, theory, data acquisition, and research methodology. Students are also introduced to current streams of research in computer science. Instructor approval required for registration.

**CSCI 6910: Master’s Thesis Research, 3 Credits**
In this course, the student conducts research under the supervision of faculty, ultimately leading to the Master’s thesis. The topics vary. Instructor approval required for admission.