COMPUTER SCIENCE AND LEARNING SCIENCES

Degree Types: PhD

The Joint PhD Program in Computer Science and Learning Sciences builds on enduring and growing connections between research on learning and computation. Rapid technological advances continue to create new and exciting ways to both understand and support learning in all settings and in all stages of life. This program is intended for students with an interest in both fields who would otherwise be forced to choose one area or the other.

Areas of Interest

The possible areas of study are broad and draw from the diverse expertise of affiliated faculty. However, all research must have clear relevance to both Computer Science and Learning Sciences. Example areas of interest include educational data mining; computational modeling as a means to understand complex scientific phenomena; adaptive technology for learning; equity issues in computing; intelligent tutoring systems; and interaction design to support learning.

Additional resources:

- Department website (https://www.sesp.northwestern.edu/learning-sciences/phd-computer-science-learning-sciences/)
- Program handbook(s)

Degree Offered

- Computer Science and Learning Sciences PhD (https://catalogs.northwestern.edu/tgs/computer-science-learning-sciences/computer-science-learning-phd/)

Learning objective(s)/Students should be able to…

- Contribute original research to the scholarly community.
- Students will have familiarity with the three stands of the learning sciences: cognition, social & cultural context, and design as well as a working understanding of a range of learning sciences research methods.
- Students will have familiarity with a breadth of computer science topic areas. Topic areas include, but are not limited to, systems, theory of computation, programming languages, algorithms, cognitive systems, human-computer interaction.
- Students will demonstrate expertise in one area of computer science within the division of graphics and interactive media or cognitive systems.

Computer Science and Learning Sciences Courses

COMP_SCI 303-0 Full Stack Software Engineering (1 Unit)
Modern software frameworks such as Next.js, Django, and Spring Boot provide the foundational components to quickly develop and deploy full-stack software applications. These frameworks, which are reusable abstractions of code wrapped in well-defined Application Programming Interfaces (APIs), provide several benefits for software developers, such as easier development and maintenance, standardization and compatibility, improved security, increased productivity, and easier collaboration. In this course we will explore one of these frameworks to create, iteratively refine, and deploy a real-world application.
Prerequisite: Students must have completed COMP_SCI 213-0 and 214-0 to enroll. CS MS and PhD students must receive instructor consent.

COMP_SCI 307-0 Introduction to Cryptography (1 Unit)
This course covers the basic knowledge in understanding and using cryptography. The main focus is on definitions, theoretical foundations, and rigorous proofs of security, with some programming practice. Topics include symmetric and public-key encryption, message integrity, hash functions, block-cipher design and analysis, number theory, and digital signatures.
Prerequisite: COMP_SCI 212 or Instructor Consent.

COMP_SCI 308-0 Foundations of Security (1 Unit)
Cybersecurity can be an intimidating word, but it doesn't need to be. It is actually fundamental to every application or system we use, however small. One does not need to be "interested" in security to be accountable for creating secure programs; anyone who will be responsible for maintaining or building any computer systems will be accountable. Have you been or do you know anyone who has been a victim of a data breach? Data breaches have become commonplace with the causes often being a simple security mistake/piece of insecure infrastructure with a massive impact. Do you see yourself building a simple application that handles user data? You would have a responsibility to protect your users' data. Are you interested in cybersecurity as a specialization or career path? You'll probably get to advise the design of systems to be secure from the ground up and mitigate security incidents in real time. And finally: do you envision yourself in a role where you will either build or manage any systems that people will use? Then your systems need to be secure. Systems, software, and tools that we use are all vulnerable to security breaches and incidents. However, security, historically and still today, is often an afterthought in the design of computer systems, which only exacerbates their vulnerability. And while cybersecurity itself is a huge and exciting field for those interested, there are core aspects of security that are accessible to and necessary for anyone in a computing-related occupation.

COMP_SCI 310-0 Scalable Software Architectures (1 Unit)
Teaches software design principles for building high-scale Internet services. Focuses on challenges arising when assembling software services that run on many machines in parallel and which require the coordination of multiple software applications.
Prerequisites: CS 213 and 214 or CS MS or CS PhDs or Instructor permission.

COMP_SCI 311-0 Inclusive Making (1 Unit)
Inclusive Making is about centering disability within computer science. The class explores the promises and shortcomings of making through a critical disability studies lens. It also looks at existing making practices within disability communities. Throughout the class, students reflect on their assumptions about disability and computer science, and wrestle with tensions related to making and accessibility alongside community organizations.
Prerequisites: COMP_SCI 110 or COMP_SCI 111 or COMP_SCI MS or COMP_SCI PhDs or Instructor consent.

COMP_SCI 312-0 Data Privacy (1 Unit)
Data breaches, privacy breaches, and concerns about algorithmic decision-making have been on the rise. As a result, data privacy has become an increasingly significant concern in the past several years. Individuals and organizations often trust institutions with their data with the expectation that one's data is private from others or to the handling institutions and that it is not used for unfair practices. To ensure the privacy of sensitive data, privacy mechanisms have been developed to

preserve the privacy of data without reducing its functionality. The goal of this course is to introduce you to the concept and implications of data privacy, including mechanisms and protocols that are used to preserve data privacy in practice. We will study concepts such as differential privacy, database anonymization, anonymous communication, and algorithmic fairness. We will also discuss privacy in the context of web privacy, social network privacy, human factors, and machine learning along with any policy implications.
Prerequisites: COMP_SCI 211 and COMP_SCI 212 and COMP_SCI 214 or COMP_SCI MS or COMP_SCI PhDs or Instructor permission - (Programming experience and familiarity with basics of discrete math and statistics/probability).

COMP_SCI 314-0 Technology and Human Interaction (1 Unit)
Understanding human interactions that occur both with and through technology; design, creation, and evaluation of technologies to support such interactions.

COMP_SCI 315-0 Design, Technology, and Research (1 Unit)
Hands-on experience in the research learning environment. Students lead research projects in social and crowd computing, cyber-learning, human-computer interaction, and artificial intelligence.
Prerequisite: consent of instructor (by application only).

COMP_SCI 320-0 Proving Properties of Programs with Mechanized Logic (1 Unit)
This course will explore the state of the art in how to implement and prove facts about software. We will focus on small, functional programs and expressing properties of them via an (extremely) sophisticated dependent type system. By the end of this course, expect to have a better understanding of how to think about properties of programs, how to write code that demonstrates that those properties are true, and some experience programming in Agda.
Prerequisite: Students must have completed COMP_SCI 212 and 321, be a MS CS Student and have completed 321, or be a CS PhD student to enroll.

COMP_SCI 321-0 Programming Languages (1 Unit)
Introduction to key parts of programming languages: syntax, semantics, and pragmatics. Implementation of a series of interpreters that show how various aspects of programming languages behave.
Prerequisites: COMP_SCI 111 and, COMP_SCI 211, and COMP_SCI 214 or COMP_SCI MS or COMP_SCI PhDs or Instructor permissiong.

COMP_SCI 322-0 Compiler Construction (1 Unit)
The compiler is the programmer's primary tool. Understanding the compiler is therefore critical for programmers, even if they never build one. Furthermore, many design techniques that emerged in the context of compilers are useful for a range of other application areas. This course introduces students to the essential elements of building a compiler: parsing, context-sensitive property checking, code linearization, register allocation, etc. To take this course, students are expected to already understand how programming languages behave, to a fairly detailed degree. The material in the course builds on that knowledge via a series of semantics preserving transformations that start with a fairly high-level programming language and culminate in machine code.
Prerequisite: COMP_SCI 213-0 and COM_SCI 214-0 or consent of instructor.

COMP_SCI 323-0 Code Analysis and Transformation (1 Unit)
Fast, highly sophisticated code analysis and code transformation tools are essential for modern software development. Before releasing its mobile apps, Facebook submits them to a tool called Infer that finds bugs by static analysis, i.e., without even having to run the code, and guides developers in fixing them. Google Chrome and Mozilla Firefox analyze and optimize JavaScript code to make browsers acceptably responsive. Performance-critical systems and application software would be impossible to build and evolve without compilers that derive highly optimized machine code from high-level source code that humans can understand. Understanding what modern code analysis and transformation techniques can and can't do is a prerequisite for research on both software engineering and computer architecture since hardware relies on software to realize its potential. In this class, you will learn the fundamentals of code analysis and transformation, and you will apply them by extending LLVM, a compiler framework now in production use by Apple, Adobe, Intel and other industrial and academic enterprises.
Prerequisites: COMP_SCI 213-0 and COM_SCI 214-0 or consent of instructor.

COMP_SCI 324-0 Dynamics of Programming Languages (1 Unit)
The goal of this course is to introduce you to the semantics of programming languages (PLs). While CS 321 explains the meaning for different PL features, such as store and control, through interpreters, this course gives them meaning using simple math. And by simple, I mean as simple as high-school algebra. Hence, the course will help you build a foundational understanding of PLs by breaking them down to their most basic ingredients without having to go through PLAI or another meta language. In other words, the course will teach you how to construct mathematical models of PLs. Moreover, because of their mathematical and foundational nature, these models will enable you to describe precisely properties of PLs that in CS 321 we only talked about in an intuitive manner.
Prerequisite: CS 321 or CS PhD or Permission Instructor.

COMP_SCI 325-0 Artificial Intelligence Programming (1 Unit)
Introduction to LISP and programming knowledge-based systems and interfaces. Strong emphasis on writing maintainable, extensible systems. Topics include semantic net-works, frames, pattern matching, deductive inference rules, case-based reasoning, and discrimination trees. Project-driven. Substantial programming assignments.
Prerequisite: COMP_SCI 110-0, COMP_SCI 111-0, or programming experience.

COMP_SCI 326-0 Introduction to the Data Science Pipeline (1 Unit)
This course aims to cover various tools in the process of data science for obtaining, cleaning, visualizing, modeling, and interpreting data. Most of the tools introduced in this course will be based on Python, although the idea can be applied to similar tools in other programming languages. As the outcome of this course, the students should be able to independently work on real-life datasets with large scales and gain insights from them.
Prerequisites: CS 212 and CS 214 or COMP_SCI MS or COMP_SCI PhDs or Instructor consent.

COMP_SCI 327-0 Generative Methods (1 Unit)
Generative Methods are algorithms which can be used to create. Programmers use grammars to make humorously chatty Twitterbots, Voronoi diagrams to create virtual cities and landscapes, and machine learned models to generate realistic oil portraits from selfies. This class will explore a range of different methods and tools, exposing you to the modern cutting edge of creative coding as you develop your own portfolio of JS/HTML apps.
Prerequisites: CS 111 and 150 or Instructor consent.

COMP_SCI 329-0 HCI Studio (1 Unit)
Human-Computer Interaction (HCI) serves as the bridge between computing and humanity. In this class we will develop our critical thinking skills by learning effective structures for designing HCI systems. We will also soften into a deeper understanding of people's problems by developing our capacities for humility, empathy, and curiosity. Learning occurs through instructional activities, team projects, and studio critique.

Prerequisite: CS 214 and has not completed CS 330 or Graduate Standing and has not completed CS 330.

COMP_SCI 330-0 Human Computer Interaction (1 Unit)
Introduction to human-computer interaction and design of systems that work for people and their organizations. Understanding the manner in which humans interact with and use computers for productive work.
For more detail - https://www.mccormick.northwestern.edu/computer-science/academics/courses/descriptions/330-1.html .
Prerequisite: CS 214 and has not completed CS329 or Graduate Standing and has not completed CS 329.

COMP_SCI 331-0 Introduction to Computational Photography (1 Unit)
Fundamentals of digital imaging and modern camera architectures. Hands-on experience acquiring, characterizing, and manipulating data captured using a modern camera platform.
Prerequisite: COMP_SCI 150 or COMP_SCI 211 or Consent of Instructor.

COMP_SCI 332-0 Online Markets (1 Unit)
Online markets are causing significant changes to society. Examples include eBay, airBnB, tinder, Uber, stackexchange, and Amazon. This class gives an introduction to the science of online markets combining topics from game theory and economics with topics from machine learning and algorithms. The two main topics of interest are how individuals in these market places optimize their strategies and how the market designer optimizes the rules of the market place so that, when individuals optimize their strategies, desired market outcomes are achieved. Student work will be a mix of problem sets and short projects.
Prerequisites: CS 212 (Discrete Math) and CS 214 (Data Structures) or CS 336 (Algorithms) or ECON 380-1 (Game Theory).

COMP_SCI 333-0 Interactive Information Visualization (1 Unit)
This course covers theory and techniques for information visualization: the use of interactive interfaces to visualize abstract data. The course targets students interested in using visualization in their work or in building better visualization tools and systems. Students will learn to design and implement effective visualizations, critique others' visualizations, conduct exploratory visual analysis, and navigate research on information visualization.
Prerequisites: COMP_SCI 214-0 or consent of instructor.

COMP_SCI 334-0 Introduction to Computational Linguistics (1 Unit)
Hands-on introduction to computational methods in empirical linguistic analysis and natural language processing. Topics include language modeling, text classification, linguistic annotation, computational semantics, and machine translation. Students will implement and apply computational models to real linguistic datasets, and conclude the course with a final project.
Prerequisite: COMP_SCI 110 or COMP_SCI 111 or be a graduate COMP_SCI student or have permission of Instructor.

COMP_SCI 335-0 Introduction to the Theory of Computation (1 Unit)
Mathematical foundations of computation, including computability, relationships of time and space, and the P vs. NP problem.
Prerequisite: COMP_SCI 212-0 or consent of instructor.

COMP_SCI 336-0 Design & Analysis of Algorithms (1 Unit)
Analysis techniques: solving recurrence equations. Algorithm design techniques: divide and conquer, the greedy method, backtracking, branch-and-bound, and dynamic programming. Sorting and selection algorithms, order statistics, heaps, and priority queues.
Prerequisite: COMP_SCI 111-0, COMP_SCI 212-0, or CS Graduate Standing or consent of instructor.

COMP_SCI 337-0 Natural Language Processing: Classical Approaches (1 Unit)
Semantics-oriented introduction to natural language processing, broadly construed. Representation of meaning and knowledge inference in story understanding, script/frame theory, plans and plan recognition, counter-planning, and thematic structures.
Prerequisite: COMP_SCI 348-0 or consent of instructor.

COMP_SCI 338-0 Practicum in Intelligent Information Systems (1 Unit)
A practical excursion into building intelligent information systems. Students develop a working program in information access, management, capture, or retrieval. Project definition, data collection, technology selection, implementation, and project management.

COMP_SCI 339-0 Introduction to Database Systems (1 Unit)
Data models and database design. Modeling the real world: structures, constraints, and operations. The entity relationship to data modeling (including network hierarchical and object-oriented), emphasis on the relational model. Use of existing database systems for the implementation of information systems.
Prerequisites: COMP_SCI 214-0 and (COMP_SCI 213-0 or COMP_ENG 205-0) or CS Graduate Standing.

COMP_SCI 340-0 Introduction to Networking (1 Unit)
A top-down exploration of networking using the five-layer model and the TCP/IP stack, covering each layer in depth. Students build web clients, servers, and a TCP implementation and implement routing algorithms.
Prerequisites: COMP_SCI 214-0 and (COMP_SCI 213-0 or COMP_ENG 205-0).

COMP_SCI 341-0 Social Networks Analysis (1 Unit)
The use of social network analysis to understand the growing connectivity and complexity in the world around us on different scales, ranging from small groups to the World 'Wide Web. How we create social, economic, and technological networks, and how they enable and constrain attitudes and behaviors. Taught with IEMS 341-0; may not receive credit for both courses.

COMP_SCI 343-0 Operating Systems (1 Unit)
Fundamental overview of operating systems, including: concurrency (processes, synchronization, semaphores, monitors, deadlock); memory management (segmentation, paging virtual memory policies); software system architectures (level structures, microkernals); file systems (directory structures, file organization, RAID); protection (access control, capabilities, encryption, signatures, authentication). Requires substantial programming projects.
Prerequisites: COMP_SCI 214-0 and COMP_SCI 213-0, or COMP_SCI 214-0 and COMP_ENG 205-0.

COMP_SCI 344-0 Design of Computer Problem Solvers (1 Unit)
Principles and practice of organizing and building artificial intelligence reasoning systems. Pattern-directed rule systems, truth-maintenance systems, and constraint languages.
Prerequisites: COMP_SCI 348-0 and COMP_SCI 325-1 or equivalent LISP experience.

COMP_SCI 345-0 Distributed Systems (1 Unit)
Basic principles behind distributed systems (collections of independent components that appear to users as a single coherent system) and main paradigms used to organize them.
Prerequisites: COMP_SCI 213-0 and COMP_SCI 214-0.

COMP_SCI 346-0 Microcontroller System Design (1 Unit)
Structure and timing of typical microprocessors. Sample microprocessor families. Memories, UARTS, timer/counters, serial devices, and related devices. MUX and related control structures for building systems. Standard bus structures. Interrupt programming. Hardware/software design tradeoffs. For syllabus, please visit - https://

www.mccormick.northwestern.edu/computer-science/academics/courses/descriptions/346.html .
Prerequisites: COMP_SCI 211 and (COMP_SCI 213 or COMP_ENG 205).

COMP_SCI 348-0 Introduction to Artificial Intelligence (1 Unit)
Core techniques and applications of AI. Representing, retrieving, and applying knowledge for problem solving. Hypothesis exploration. Theorem proving. Vision and neural networks.
Prerequisites: COMP_SCI 111 and COMP_SCI 214 or COMP_SCI 111 and CogSci major or COMP_SCI MS or COMP_SCI PhDs or Instructor permission.

COMP_SCI 349-0 Machine Learning (1 Unit)
Study of algorithms that improve through experience. Topics typically include Bayesian learning, decision trees, genetic algorithms, neural networks, Markov models, and reinforcement learning. Assignments include programming projects and written work.
Prerequisites: COMP_SCI grad standing OR (COMP_SCI 214 and (MATH 240-0 or GEN_ENG 205-1 or GEN_ENG 206-1) and (IEMS 201-0 or IEMS 303-0 or ELEC_ENG 302-0 or STAT 210-0 or MATH 310-1).

COMP_SCI 350-0 Introduction to Computer Security (1 Unit)
Basic principles and practices of computer and information security. Software, operating system, and network security techniques, with detailed analysis of real-world examples. Topics include cryptography, authentication, software and operating system security (e.g., buffer overflow), Internet vulnerability (DoS attacks, viruses/worms, etc.), intrusion detection systems, firewalls, VPN, and web and wireless security.
Prerequisite: COMP_SCI 213-0 or equivalent or consent of instructor; COMP_SCI 340-0 highly recommended.

COMP_SCI 351-1 Introduction to Computer Graphics (1 Unit)
Mathematical software and hardware requirements for computer graphics systems. Data structures and programming languages. Random displays. Graphic applications.
Prerequisite: COMP_SCI 214-0 or Graduate standing.

COMP_SCI 351-2 Intermediate Computer Graphics (1 Unit)
Methods and theory of computer graphics. Project-oriented approach. Describing shapes, movement, and lighting effects; interactive elements.
Prerequisites: COMP_SCI 214-0 and COMP_SCI 351-1 or Graduate standing.

COMP_SCI 352-0 Machine Perception of Music & Audio (1 Unit)
Machine extraction of musical structure in audio and MIDI and score files, covering areas such as source separation and perceptual mapping of audio to machine-quantifiable measures.
Prerequisite: COMP_SCI 211-0 and COMP_SCI 214-0.

COMP_SCI 353-0 Natural & Artificial Vision (1 Unit)
This course covers the mathematical operations underlying computer vision and their embodiment in naturally-occurring visual systems. Students will learn basics of image processing, neuroscience, and computational imaging.

COMP_SCI 354-0 Computer System Security (1 Unit)
The past decade has seen an explosion in the concern for the security of information. This course introduces students to the basic principles and practices of computer system and networking security, with detailed analysis of real-world examples and hands-on practice. Topics include the basic crypto, authentication, reverse engineering, buffer overflow attacks, vulnerability scanning, web attacks, firewalls, intrusion detection/prevention systems, etc. We will first introduce the basic theory for each type of attack; then we will actually carry them out in 'real-world' settings. The goal is to learn security by learning how to view your machine from a hacker's perspective. In addition, we encourage students to participate in the UCSB International Capture the Flag Competition. Capture the Flag is a network security exercise where the goal is to exploit other machines while defending your own. In fact, this course should prepare you for any one of many capture the flag competitions that take place year-round. We will learn about different types of hacks and perform them. After learning how to execute such exploits and penetrate a network, we will discuss ways to protect a network from others exploiting the same vulnerabilities. Understanding security is essential in all fields of software development and computing. For major or minors in Computer Science, this course can satisfy the system breadth.
Prerequisite: COMP_SCI 211-0 and COMP_SCI 213-0 or COMP_SCI 211-0 and COMP_ENG 205-0.

COMP_SCI 355-0 Digital Forensics and Incident Response (1 Unit)
This course aims to teach students the concepts of Digital Forensics and Incident Response. The technical content taught in the class consists of deep knowledge of filesystems and operating systems so that students know which digital artifacts to investigate in data breach scenarios. Labs and assignments are a sanitized version of real-world intrusions by nation-state actors and cybercriminals.
Prerequisites: COMP_SCI 213-0 and COMP_SCI 354-0 or COMP_SCI 213-0 and COMP_SCI 343-0.

COMP_SCI 362-0 Foundations of Quantum Computing and Quantum Information (1 Unit)
This course will be an introduction to the theory of quantum computation from a computer science perspective. Quantum computing holds great promise for obtaining substantial computational improvements over classical computing for many problems. In this course, we will cover the basics of quantum computation, and different topics that explore both the capabilities and limitations of quantum computers. Topics will include (subject to change) the basics of quantum information, quantum circuits, quantum algorithms, quantum complexity theory, quantum query complexity, and quantum communication complexity. No knowledge of quantum mechanics is required. We will cover the necessary physics concepts that are needed for this course.

COMP_SCI 367-0 Wireless and Mobile Health: Passive Sensing Data Analytics (1 Unit)
A hands-on introduction and experience to the growing field of mobile Health. Students work together on a project with clinicians and faculty in medicine, building a unique mHealth system while testing their system on a small population. Theory-driven project hypothesis, technology selection and development, passive sensing data analytic chain understanding and implementation, and project management.

COMP_SCI 368-0 Programming Massively Parallel Processors with CUDA (1 Unit)
This course focuses on developing and optimizing applications software on massively parallel graphics processing units (GPUs). Such processing units routinely come with hundreds to thousands of cores per chip and sell for a few hundred to a few thousand dollars. The massive parallelism they offer allows applications to run 2x-450x faster than on conventional multicores. However, to reach this performance improvement, the application must fully utilize the computational, storage and communication resources provided by the device. This course discusses state-of-the-art parallel programming optimization methods to achieve this goal.
Prerequisites: COMP_SCI 213 and (COMP_SCI 211 or COMP_SCI 230) or consent of instructor.

COMP_SCI 370-0 Computer Game Design (1 Unit)
Plot, narrative, and character simulation for creating game worlds; artificial intelligence for synthetic characters; tuning gameplay. Substantial programming and project work.

Prerequisites: COMP_SCI 214-0; 1 unit of COMP_SCI 322-0, COMP_SCI 343-0, COMP_SCI 348-0, or COMP_SCI 351-1, COMP_SCI 351-2.

COMP_SCI 371-0 Knowledge Representation and Reasoning (1 Unit)
Principles and practices of knowledge representation, including logics, ontologies, commonsense knowledge, and semantic web technologies.
Prerequisites: COMP_SCI 348-0 or COMP_SCI 325, or equivalent experience with artificial intelligence.

COMP_SCI 374-0 Causal Graphical Models (1 Unit)
We know that correlation does not imply causation, but careful analyses of correlations are often our only way to quantify cause and effect in domains ranging from healthcare to education. This courses introduces causal inference methods, primarily using probabilistic graphical models, to identify and estimate counterfactual quantities as functions of observational data. We will discuss common challenges to causal inference, including confounding bias, missing data, measurement error, and selection bias. The final project will allow students to choose a dataset on which to perform a causal data analysis.

COMP_SCI 376-0 Computer Game Design and Development (1 Unit)
Introduction to design of simulation-based media, with an emphasis on 2D game design. Mathematical preliminaries: linear, affine, and projective spaces, linear transforms, inner and exterior products, unit quaternions; Architecture: update/render loop, component systems, serialization and deserialization, event handling and asynchronous processing, multitasking; Rendering: scene graphs, meshes, shaders, sprites; Networking; Audio; Physics: particles, rigid bodies, collision detection; Gameplay design.
Prerequisite: COMP_SCI 214-0.

COMP_SCI 377-0 Game Design Studio (1 Unit)
In this course, students will design and develop games using the Unity game engine, with focus on team-based projects and agile development practices. Lectures will cover game design theory, game architecture and implementation, and the business of game development. Students will participate in class discussion and evaluation of projects in progress, to develop their skills in iterative design and implementation.
Prerequisite: COMP_SCI 214 and COMP_SCI 376-0.

COMP_SCI 388-0 Software Engineering Beyond Programming (1 Unit)
Many important issues and tradeoffs in software engineering appear only once projects reach a certain scale: large codebases, large teams, long periods of time. In this class, we will study a number of such issues, as well as practices and processes to help manage them.
Registration is By Application Only

COMP_SCI 392-0 Rapid Prototyping for Software Innovation (1 Unit)
This is a course about developing working prototypes of full-stack mobile web software applications in rapid iterations. Teams design and implement three distinct applications over ten weeks. These projects are the context for introducing (1) cross-functional team development, (2) lean agile value-first product development, and (3) specific web application frameworks and development tools, such as React, Firebase, Cypress, and Github Actions for continuous integration.
Prerequisites: CS junior, senior, or graduate OR permission of the instructor. Familiarity with JavaScript and HTML is presumed.

COMP_SCI 393-0 Software Construction (1 Unit)
Building software is a craft that requires careful design. This course teaches software design principles in a studio setting. Each week, students present their programs to the class for review. Together, the class evaluates the programs for correctness and, more importantly, clarity and design. Expect to learn how to build reliable, maintainable, extensible software and how to read others' codes.
Prerequisites: COMP_SCI 211-0 and COMP_SCI 214-0.

COMP_SCI 394-0 Agile Software Development (1 Unit)
Developing mobile and web applications, using modern sustainable agile practices, such as backlogs, user stories, velocity charts, and test driven development, to deliver value as quickly as possible to end users, clients, developers, and the development organization.
Prerequisites: Consent of instructor.

COMP_SCI 396-0 Special Topics in Computer Science (1 Unit)
Topics suggested by faculty and approved by the department. Equivalent to 397 but intended to apply toward courses for the computer science major.

COMP_SCI 397-0 Special Projects in Computer Science (1 Unit)
Projects suggested by faculty and approved by the department. Equivalent to 396 but intended to apply toward courses for the computer science major and its project requirement.

COMP_SCI 401-0 Introduction to Graduate Studies (1 Unit)
An introduction to graduate studies in Computer Science aimed at preparing incoming doctoral students to perform great research, regardless of area, and providing an overview of research areas in computer science at Northwestern through short presentations by faculty.

COMP_SCI 407-0 Advanced Topics in Modern Cryptography (1 Unit)
Large-scale computing environments nowadays bring great opportunities in terms of the ubiquity and omnipresence of data, communication, and computation. However, such opportunities often come with concerns on trust and security when exchanging data and computing on joint data. Modern cryptography features precise definitions and rigorous proofs of security for complicated protocols. It is a cornerstone for many large-scale systems used now and crucial for newly emerging systems like e-voting and blockchain. This course is a graduate-level introduction to modern cryptography. The course will cover definitions, constructions, and proofs used in modern cryptography. Topics include secure multi-party computation, zero-knowledge proofs, oblivious computation, fully homomorphic encryption, decentralized cryptocurrencies, and blockchain applications. Various techniques and assumptions will be introduced in the course, including simulation-based security, Diffie Hellman (DH) assumption, Learning With Error (LWE) assumption, etc.

COMP_SCI 409-0 Swarms and Multi-Robot Systems (1 Unit)
This class surveys the state of the art research in robotic swarms, looking at both algorithms for controlling them and current hardware implementations. It also addresses the deficiencies keeping them from every-day use. Coursework includes reading research papers, student presentations and discussion of select papers, and projects implementing studied topics in a real or simulated robot swarm.
Prerequisite: Permission of Instructor. Cross-listed with MECH_ENG 409-0.

COMP_SCI 410-0 Autonomous Quadrotor Design and Control (1 Unit)
Centered around a project where teams create and program an autonomous quadrotor robot, this class focuses on advanced embedded control of an electromechanical system. Topics include: programming interfaces between an embedded computer and external sensors/actuators, programming a timing-critical control loop for stable flight, and creating a software stack that interacts with low-level code to create a desired high-level behavior.
Prerequisite: Permission of Instructor. Cross-listed with MECH_ENG 410-0.

COMP_SCI 412-0 Data Privacy (1 Unit)
Data breaches, privacy breaches, and concerns about algorithmic decision-making have been on the rise. As a result, data privacy has

become an increasingly significant concern in the past several years. Individuals and organizations often trust institutions with their data with the expectation that one's data is private from others or to the handling institutions and that it is not used for unfair practices. To ensure the privacy of sensitive data, privacy mechanisms have been developed to preserve the privacy of data without reducing its functionality. The goal of this course is to introduce you to the concept and implications of data privacy, including mechanisms and protocols that are used to preserve data privacy in practice. We will study concepts such as differential privacy, database anonymization, anonymous communication, and algorithmic fairness. We will also discuss privacy in the context of web privacy, social network privacy, human factors, and machine learning along with any policy implications.
Prerequisites: COMP_SCI 211 and COMP_SCI 212 and COMP_SCI 214 or COMP_SCI MS or COMP_SCI PhDs or Instructor permission - (Programming experience and familiarity with basics of discrete math and statistics/probability).

COMP_SCI 413-0 Tangible Interaction Design and Learning (1 Unit)
Explores the use of tangible interaction to create innovative learning experiences, including distributed cognition, embodied interaction, cultural forms, and design frameworks.
Prereq: COMP_SCI 110-0 or COMP_SCI 111-0.

COMP_SCI 415-0 Design, Technology, and Research (1 Unit)
Research learning environment to realize and develop students' potential for developing novel technologies and creative solutions through design, engineering, and research. Students lead research projects in social and crowd computing, cyberlearning, human computer interaction, and artificial intelligence.
Prerequisites: consent of instructor (by application only).

COMP_SCI 424-0 Dynamics of Programming Languages (1 Unit)
The goal of this course is to introduce you to the semantics of programming languages (PLs). While CS 321 explains the meaning for different PL features, such as store and control, through interpreters, this course gives them meaning using simple math. And by simple, I mean as simple as high-school algebra. Hence, the course will help you build a foundational understanding of PLs by breaking them down to their most basic ingredients without having to go through PLAI or another meta language. In other words, the course will teach you how to construct mathematical models of PLs. Moreover, because of their mathematical and foundational nature, these models will enable you to describe precisely properties of PLs that in CS 321 we only talked about in an intuitive manner.
Prerequisite: CS 321 or CS PhD or Permission Instructor.

COMP_SCI 430-0 Design of Interactive Learning Environments (1 Unit)
Design of computer-based "learning-by-doing" environments. Course focuses more on initial conception of learning environments than on technical issues involved in building these environments.

COMP_SCI 433-0 Wireless Protocols for the Internet of Things (1 Unit)
The Internet of Things promises a world of computers woven into our physical world. A common need for these devices is low-power, wireless communication. The goal of this course is to introduce students to a variety of wireless networks that target low-power, machine-to-machine communication as is common in the Internet of Things. While we introduce the physical layer and have a goal of getting data to the internet at large, the focus of this course is on the wireless protocols themselves. How are packets structured, and why? How are they designed to enable low-power communication? How do they deal with contention and reliability? What makes them more or less suitable for different applications? We will explore local-area protocols such as Bluetooth Low Energy, Thread (and other 802.15.4 protocols), low-power, wide-area networks (LPWANs) such as LoRaWAN and Sigfox, and other related topics such as backscatter and localization. The class will include lectures on these topics as well as practical hands-on lab sessions interacting with networks. Starting very early in the class, students will form groups for their final project and create a proposal, allowing them at least 5 weeks for project work. Projects will culminate in a research-paper-style write-up and a class presentation.
Prerequisite: CS 346 or CS 340 or CE 346 or EE 326.

COMP_SCI 436-0 Graduate Algorithms (1 Unit)
This is an introductory graduate level course on Algorithms that will give broad exposure to recent advances in algorithms, yet cover the fundamental techniques needed to understand the recent advances in algorithms research. At the end of the course, students will be able to read and understand research papers in most recent areas of algorithms research.

COMP_SCI 437-0 Approximation Algorithms (1 Unit)
This course studies approximation algorithms – algorithms that are used for solving hard optimization problems. Such algorithms find approximate (slightly suboptimal) solutions to optimization problems in polynomial time. Unlike heuristics, approximation algorithms have provable performance guarantees: they have bounds on the running time and on the quality of the obtained solutions. In this course, we will introduce various algorithmic techniques used for solving optimization problems such as greedy algorithms, local search, dynamic programming, linear programming (LP), semidefinite programming (SDP), LP duality, randomized rounding, and primal-dual analysis. The course assumes background in basic probability theory and discrete mathematics. Key mathematical concepts will be reviewed before they are used.
Prerequisistes: COMP_SCI 212 and COMP_SCI 336 (or similar courses) or CS MS or CS PhDs.

COMP_SCI 440-0 Advanced Networking (1 Unit)
This course will cover a broad range of topics including Internet evolution and architectures; analysis and design of network protocols (both wired and wireless); networking issues for Web and gaming applications; analysis and performance of content distribution networks; network security, vulnerability, and defenses.
Prerequisites: COMP_SCI 340-0 or permission of instructor.

COMP_SCI 441-0 Resource Virtualization (1 Unit)
The bulk of the time in this class examining a virtual machine monitor (VMM) in depth, at the source code level. The course explains the hardware/software interface of a modern x86 computer in detail. A VMM is an operating system that is implemented directly on top of the hardware interface, and itself presents a hardware interface to higher-level software. Students will also acquire valuable kernel development skills.
Prerequisites: COMP_SCI 213-0.

COMP_SCI 443-0 Advanced Operating Systems (1 Unit)
Advanced concepts in operating systems and distributed computing from historical perspectives to current themes such as peer-to-peer computing and mobile systems.

COMP_SCI 445-0 Internet-scale Experimentation (1 Unit)
Internet-scale Experimentation is a graduate-level seminar exploring the challenges of large-scale networked system experimentation and measurements. Per-Requisite: Students must have taken CS 340-0 or CS 345-0 or CS Graduate standing or Consent of Instructor.

COMP_SCI 446-0 Kernel and Other Low-level Software Development (1 Unit)
The development of low-level systems software such as drivers, kernels, etc is very different from the development of applications. This class

teaches how such development is done: how to design, implement, debug, and optimize low-level software and use available tools.
Prerequisites: (COMP_SCI 213-0 or COMP_ENG 205-0) and (COMP_SCI 343-0 or COMP_ENG 361-0 or COMP_ENG 366-0 or COMP_ENG 466-0) or permission of Instructor.

COMP_SCI 447-0 Conversational AI (1 Unit)
Principles and practices of creating AI systems which interact with people through conversations. This includes knowledge-rich natural language understanding, multimodal interactions (i.e. speech and sketching), principles of dialogue drawn from cognitive science, question-answering, and architectures for building conversational AI systems. Involves substantial programming and project work. Class sessions include both lectures and studio instruction.
Prerequisites: COMP_SCI 371 or permission of instructor.

COMP_SCI 449-0 Deep Learning (1 Unit)
Deep learning is a branch of machine learning based on algorithms that try to model high-level abstract representations of data by using multiple processing layers with complex structures. Some representations make it easier to learn tasks (e.g., face recognition or spoken word recognition) from examples. One of the promises of deep learning is replacing handcrafted features with efficient algorithms for unsupervised or semi-supervised feature learning and hierarchical feature extraction. In this course students will study deep learning architectures such as autoencoders, , convolutional deep neural networks, and recurrent neural networks. They will read original research papers that describe the algorithms and how they have been applied to fields like computer vision, automatic speech recognition, and audio event recognition. For projects, students can work on their own or in groups (recommended) to write a codebase that reproduces a landmark research paper. This course is aimed at advanced undergraduates, masters, and PhD students.
Prerequisites: CS 349 or CS PhD or Instructor permission.

COMP_SCI 450-0 Internet Security (1 Unit)
Through measurement-based approaches, students analyze the complexity of the Internet, and develop countermeasures against various vulnerabilities of the Internet such as viruses, worms, and denial of service attacks.
Prerequisites: Comp_sci 340 or Comp_sci 343 or Comp_sci 354 or Instructor permission.

COMP_SCI 455-0 Distributed Computing Systems (1 Unit)
Fundamentals and systems design aspects of distributed systems, paradigms for distributed computing, client-server computing, concurrency control, distributed file systems, resource management, high-performance computing aspects.
Prerequisites: COMP_SCI 343-0 or COMP_ENG 361-0.

COMP_SCI 457-0 Advanced Algorithms (1 Unit)
Analysis and design of algorithms; amortized analysis; arithmetic circuits; computational geometry; NP-completeness; approximation algorithms.
Prerequisites: COMP_SCI 336-0 or any algorithms course.

COMP_SCI 460-0 Introduction to Law and Digital Technologies (1 Unit)
This course explores the legal implications of the contemporary technology landscape, including the growth of artificial intelligence, the ecosystem for creating and disseminating digital information, and the challenges of ensuring digital privacy and algorithm equity. The course aims to help students develop a broad, contextualized view of the legal and policy opportunities and challenges associated with rapid technological change. A key goal of the course is for students to acquire the skills to understand, contribute to, and shape the dialog on complex issues at the intersection of technology and law.
Prerequisites: Graduate standing and an interest and willingness to learn.

COMP_SCI 461-0 Deep Learning for Natural Language Processing (1 Unit)
In the first half of this course, we will explore the evolution of deep neural network language models, starting with n-gram models and proceeding through feed-forward neural networks, recurrent neural networks and transformer-based models. In the second half of the course we will apply these models to natural language processing tasks, including question answering, text classification (including fakes detection), text summarization, text generation (including dialogue, neural machine translation and program synthesis) and natural language inference, among others. After completing this course, students will be able to generalize these techniques to a wide variety of applied and research problems in natural language processing.

COMP_SCI 462-0 Foundations of Quantum Computing and Quantum Information (1 Unit)
This course will be an introduction to the theory of quantum computation from a computer science perspective. Quantum computing holds great promise for obtaining substantial computational improvements over classical computing for many problems. In this course, we will cover the basics of quantum computation, and different topics that explore both the capabilities and limitations of quantum computers. Topics will include (subject to change) the basics of quantum information, quantum circuits, quantum algorithms, quantum complexity theory, quantum query complexity, and quantum communication complexity. No knowledge of quantum mechanics is required. We will cover the necessary physics concepts that are needed for this course.

COMP_SCI 468-0 Programming Massively Parallel Processors with CUDA (1 Unit)
This course focuses on developing and optimizing applications software on massively parallel graphics processing units (GPUs). Such processing units routinely come with hundreds to thousands of cores per chip and sell for a few hundred to a few thousand dollars. The massive parallelism they offer allows applications to run 2x-450x faster than on conventional multicores. However, to reach this performance improvement, the application must fully utilize the computational, storage and communication resources provided by the device. This course discusses state-of-the-art parallel programming optimization methods to achieve this goal.
Prerequisites: COMP_SCI 213 and (COMP_SCI 211 or COMP_SCI 230) or consent of instructor.

COMP_SCI 469-0 Machine Learning & Artificial Intelligence for Robotics (1 Unit)
A coverage of artificial intelligence, machine learning and statistical estimation topics that are especially relevant for robot operation and robotics research. Prerequisite: Graduate-level standing (or permission of instructor) for the maths. Some programming experience (in Matlab okay). Undergraduate student enrollment by permission of instructor only.
Prerequisites: CS PhD, MECH_ENG PhD, MSR Students or by permission only.

COMP_SCI 472-0 Designing and Constructing Models with Multi-Agent Languages (1 Unit)
This course will begin with an introduction to the multi-agent language NetLogo. Students will design and implement several NetLogo models and analyze their behavioral regimes. Students will also learn to build models of interaction on social networks (or other types of networks). We will cover methodology for verifying, validating and replicating agent-based models and comparisons with systems dynamics and equation-based models. NetLogo comes with many extensions that support a

variety of additional features. Students can use these extensions to create specialized models, such as complex networks, real-time data extraction, data mining, connections to physical devices, etc.. Students will also have an opportunity to explore existing and create their own participatory simulations using the HubNet architecture as well as exploring connecting real world sensors and motors to models. Students can also explore multi-level agent-based modeling in which hundreds or thousands of models are connected with NetLogo's LevelSpace extension.
Prerequisites: Comp_sci PhDs or Comp_Eng PhDs or TSB PhDs or CS LS PhDs or Consent of instructor.

COMP_SCI 474-0 Probabilistic Graphical Models (1 Unit)
Probabilistic graphical models are a powerful technique for handling uncertainty in machine learning. The course will cover how probability distributions can be represented in graphical models, how inference and learning are performed in the models, and how the models are utilized for machine learning in practice.
Prerequisites: COMP_SCI 349 or permission of the instructor.

COMP_SCI 496-0 Special Topics in Computer Science (1 Unit)
Topics suggested by faculty and approved by the department.

COMP_SCI 497-0 Special Projects in Computer Science (1 Unit)
Project-based course on topics suggested by faculty and approved by the department.

COMP_SCI 499-0 Projects (1 Unit)
Special projects carried out under faculty direction. Permission of instructor and department required.

COMP_SCI 510-0 Seminar (1 Unit)
Seminar on topics of current interest.

COMP_SCI 590-0 Research (1-4 Units)
Independent investigation of selected problems pertaining to thesis or dissertation. May be repeated for credit.

LRN_SCI 401-0 Knowledge Representation for the Learning Sciences (1 Unit)
Theoretical and methodological techniques for knowledge representation, primarily as practiced by cognitive scientists. Application of these techniques to issues of learning that are of interest to researchers in the learning sciences.

LRN_SCI 402-0 Social Dimensions of Teaching & Learning (1 Unit)
Students' relationships with one another and with teachers in school and nonschool settings. Implications for classroom instruction of social learning theory, student diversity, classroom climate, cooperative and competitive goal structures, and processes of attribution and achievement motivation.

LRN_SCI 403-0 Foundations of the Learning Science (1 Unit)
Cognitive and social science theories of how people learn to understand, reason, and solve problems. Implications for the design of classroom learning environments; learning in real scenarios for investigating central issues in cognitive science. Learning in mathematics, science, reading/writing, and informal reasoning.

LRN_SCI 404-0 Methods and Epistemologies for the Study of Learning 1 (1 Unit)
This course focuses on the development of research questions and understanding the range of possible methodological approaches to understanding learning based on those questions.

LRN_SCI 405-1 Methods and Epistemologies for the Study of Learning II (1 Unit)

LRN_SCI 410-0 Quantitative Methods I: Probability and Statistics (1 Unit)

LRN_SCI 411-0 Quantitative Methods II: Regression Analysis (1 Unit)

LRN_SCI 413-0 Tangible Interaction Design and Learning (1 Unit)
This course will explore the use of tangible interaction to create innovative learning experiences. It will review both theoretical and technological foundations of the field. Topics include creative expression, embodied interaction, cultural forms, and design frameworks.

LRN_SCI 415-0 Field Methods (1 Unit)
The purpose of this course is to introduce students to the world of qualitative research so that they will be able to read qualitative studies intelligently, and learn to design and conduct qualitatively oriented studies themselves. Beginning with an overview of the epistemological assumptions behind different kinds of research, the course will explore various types of qualitative research approaches and the kinds of topics and queries they support. Students will read and critique examples of published research of various kinds. Next, students will investigate the various methods of collecting qualitative data.#The class is designed so that students simultaneously read about and discuss qualitative research, and gather data themselves.#Although the course touches on analysis, the main focus is on developing a qualitative research project and collecting data for it.

LRN_SCI 416-0 Advanced Qualitative Methods (1 Unit)

LRN_SCI 425-0 Introduction to Design for the Learning Sciences (1 Unit)
Building the skills and knowledge necessary to support the design of educational experiences. Exploration of general design principles and learning sciences theoretical perspectives through examination of existing cases of instructional design. A design project involving needs analysis, specifying learning objectives, and designing a new educational experience.

LRN_SCI 426-0 Design of Technological Tools for Thinking and Learning (1 Unit)

LRN_SCI 429-0 Design of Learning Environments (1 Unit)
Issues in designing and studying innovative learning environments. New models of classroom interaction, particularly using technology to enable new cognitive and social roles for students. Topics include simulations, tutors, computer-mediated communication, project-based learning. Theoretical motivations in cognitive and social-interaction learning theories, empirical studies evaluating their effectiveness, and prospects for propagation of such innovations.

LRN_SCI 434-0 Teacher Thinking & Learning (1 Unit)
Recent research on teacher cognition, how teacher knowledge is organized and accessed, and relationship between knowledge and practices. Investigate novice and veteran teachers learning.

LRN_SCI 438-0 Teaching with Technology (1 Unit)
Conceptual strategies for integrating technology into effective pedagogy and practical strategies for employing technology in classrooms. Includes hands-on experience with technology and a design project.

LRN_SCI 442-0 Social Policymaking and Policy Implementation (1 Unit)

LRN_SCI 443-0 Educational Policy: Design, Implementation and Effects (1 Unit)
Introduction to issues in educational reform. Analyzing educational reform; framing educational policy problems; examining reformers' assumptions about the school system, about the roles of school in society, and about teaching and learning. The course is grounded in school decentralization, systemic reform and school choice.

LRN_SCI 451-0 Topics in Learning Sciences (1 Unit)

Discussion of trends in the field of Learning Sciences via articles and other resources.

LRN_SCI 452-0 Constructionism Seminar (1 Unit)
Discussion of trends in the field of Learning Sciences via articles and other resources.

LRN_SCI 463-0 Topics in Research Methods (1 Unit)
Methodological approaches to research on learning- teaching environment implementation. Methods for examining processes of change and adoption of educational interventions in various settings. May be repeated for credit with change of topic.

LRN_SCI 472-0 Designing and Constructing Models With Multi-Agent Languages (1 Unit)
Contact the department for further information.

LRN_SCI 477-0 Philosophical & Historical Foundations of Education Reform (1 Unit)
How influential root metaphors for the learner, knowledge, and learning processes become embodied in educational technologies, and how the sociocultural context of their design and use influences their appropriation or rejection.

LRN_SCI 499-0 Independent Study (1-3 Units)
SEE DEPT FOR SECTION AND PERMISSION NUMBERS.

LRN_SCI 519-0 Responsible Conduct of Research Training (0 Unit)

LRN_SCI 590-0 Research (1-3 Units)
Independent investigation of selected problems pertaining to thesis or dissertation. May be repeated for credit. - SEE DEPT FOR SECTION AND PERMISSION NUMBERS.