https://csls.sesp.northwestern.edu/

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.

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

**Computer Science and Learning Sciences Courses**

**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 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 111-0, COMP_SCI 214-0.

**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.

**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. Prerequisite: COMP_SCI 213-0.

**COMP_SCI 325-1 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 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. Prerequisite: programming experience.

**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.

**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 consent of instructor.

**COMP_SCI 337-0 Natural Language Processing (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).

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 Mechanism Design (1 Unit)
Applying algorithms and microeconomics to derive a theory of the design of mechanisms that produce desired outcomes despite counteractive inputs by outside agents. Key application areas: auctions, markets, networking protocols.

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, microkernels); file systems (directory structures, file organization, RAID); protection (access control, capabilities, encryption, signatures, authentication). Requires substantial programming projects.

Prerequisites: Both 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 348-0 Introduction to Artificial Intelligence (1 Unit)

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.

Prerequisite: COMP_SCI 348-0.

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 351-0 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.

COMP_SCI 351-1 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.

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, GEN_ENG 205-2, or prior programming experience in MATLAB.

COMP_SCI 354-0 Network Penetration & Security (1 Unit)
Practical tools for vulnerability assessment and defense of computer and communication systems.

Prerequisites: COMP_SCI 213-0 or COMP_ENG 205-0; ELEC_ENG 333-0 or COMP_SCI 340-0.

COMP_SCI 357-0 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 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 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 348-0, COMP_SCI 325-1, or equivalent experience with artificial intelligence.

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 376-0.

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,
testable software and to how to read others’ codes.
Prerequisites: COMP_SCI 111-0 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.

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

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

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 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 431-0 Human Perception and Electronic Media (1 Unit)

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)
Explores the challenges of large-scale networked system experimentation
and measurement.
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 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 450-0 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 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.