

# COMPUTER AND INFORMATION SCIENCE (CIS)

## CIS 0099 Undergraduate Research/Independent Study

An opportunity for the student to become closely associated with a professor (1) in a research effort to develop research skills and techniques and/or (2) to develop a program of independent in-depth study in a subject area in which the professor and student have a common interest. The challenge of the task undertaken must be consistent with the student's academic level. To register for this course, the student must submit a detailed proposal, signed by the independent study supervisor, to the SEAS Office of Academic Programs (111 Towne) no later than the end of the "add" period. Prerequisite: A maximum of 2 c.u. of CIS 0099 may be applied toward the B.A.S. or B.S.E. degree requirements.

Fall or Spring  
1 Course Unit

## CIS 1050 Computational Data Exploration

The primary goal of this course is to introduce computational methods of interacting with data. In this course, students will be introduced to the IPython programming environment. They will learn how to gather data, store it in appropriate data structures and then either write their own functions or use libraries to analyze and then display the salient information in that data. Data will be drawn from a variety of domains, including but not limited to travel, entertainment, politics, economics, biology etc.

Fall or Spring

Mutually Exclusive: PHYS 1100

1 Course Unit

## CIS 1070 Visual Culture through the Computer's Eye

Visual studies and the humanities more generally have thought about and modeled seeing of artworks for many centuries. What useful tools can machine learning develop from databases of art historical images or other datasets of visual culture? Can tools from machine learning help visual studies ask new questions? When put together, what can these fields teach us about visual learning, its pathways, its underlying assumptions, and the effects of its archives/datasets? Class project teams will ideally be composed of both humanities majors and engineering majors who will develop datasets and/or ask important questions of datasets, in addition to thinking and writing more generally about how computer vision could help in teaching and analyzing visual art. We are looking for a variety of students from different majors and schools to bring their diverse skill sets to the course. No programming knowledge is required. The course offers an example-based introduction to machine learning, so no prior knowledge of machine learning is required.

Fall or Spring

Also Offered As: VLST 2090

1 Course Unit

## CIS 1100 Introduction to Computer Programming

Introduction to Computer Programming is the first course in our series introducing students to computer science. In this class you will learn the fundamentals of computer programming, with emphasis on applications in science and engineering. You will also learn about the broader field of computer science and algorithmic thinking, the fundamental approach that computer scientists take to solving problems.

Fall or Spring

1 Course Unit

## CIS 1200 Programming Languages and Techniques I

A fast-paced introduction to the fundamental concepts of programming and software design. This course assumes some previous programming experience, at the level of a high school computer science class or CIS 1100. (If you got at least 4 in the AP Computer Science A or AB exam, you will do great.) No specific programming language background is assumed: basic experience with any language (for instance Java, C, C++, VB, Python, Perl, or Scheme) is fine. If you have never programmed before, you should take CIS 1100 first.

Fall or Spring

1 Course Unit

## CIS 1210 Programming Languages and Techniques II

This is a course about Algorithms and Data Structures using the Java programming language. We introduce the basic concepts about complexity of an algorithm and methods on how to compute the running time of algorithms. Then, we describe data structures like stacks, queues, maps, trees, and graphs, and we construct efficient algorithms based on these representations. The course builds upon existing implementations of basic data structures in Java and extends them for the structures like trees, studying the performance of operations on such structures, and their efficiency when used in real-world applications.

Spring

Prerequisite: CIS 1200 AND CIS 1600

1 Course Unit

## CIS 1250 Technology and Policy

Have you ever wondered why sharing music and video generates such political and legal controversies? Is information on your PC safe and should law enforcement be able to access information you enter on the Web? Will new devices allow tracking of your every move and every purchase? CIS 1250 is focused on developing an understanding of existing and emerging technologies, along with the political, societal and economic impacts of those technologies. The technologies are spread across a number of engineering areas and each of them raise issues that are of current concern or are likely to be a future issue.

1 Course Unit

## CIS 1400 Introduction to Cognitive Science

How do minds work? This course surveys a wide range of answers to this question from disciplines ranging from philosophy to neuroscience. The course devotes special attention to the use of simple computational and mathematical models. Topics include perception, learning, memory, decision making, emotion and consciousness. The course shows how the different views from the parent disciplines interact and identifies some common themes among the theories that have been proposed. The course pays particular attention to the distinctive role of computation in such theories and provides an introduction to some of the main directions of current research in the field. It is a requirement for the BA in Cognitive Science, the BAS in Computer and Cognitive Science, and the minor in Cognitive Science, and it is recommended for students taking the dual degree in Computer and Cognitive Science.

Fall

Also Offered As: COGS 1001, LING 1005, PHIL 1840, PSYC 1333

1 Course Unit

## CIS 1600 Mathematical Foundations of Computer Science

What are the basic mathematical concepts and techniques needed in computer science? This course provides an introduction to proof principles and logics, functions and relations, induction principles, combinatorics and graph theory, as well as a rigorous grounding in writing and reading mathematical proofs.

Spring

1 Course Unit

**CIS 1810 The Quantum and the Computer**

This Freshman Seminar is designed to be a very introductory exposition about Quantum Computation and Quantum Information Science. There are no formal physics, mathematics, or computer science prerequisites. It is meant primarily for freshmen in SAS and Wharton, who have an itch to learn about a beautiful subject that intrinsically unites quantum physics, computation, and information science. The structure of the course will be lecture-based using small-team based exercises for evaluation. The enrollment will be limited to 20 students. Freshmen standing.

Fall or Spring  
1 Course Unit

**CIS 1901 C++ Programming**

This course will provide an introduction to programming in C++ and is intended for students who are already experienced with programming in C and in object-oriented languages such as Java. C++ provides programmers with a greater level of control over machine resources and is commonly used in situations where low level access or performance are important. This course will cover the features and abstractions that C++ provides to write code that is both safe and performant. This course recommends students to have completed CIS 1200 and CIS 2400.

Not Offered Every Year  
0-0.5 Course Units

**CIS 1902 Python Programming**

Python is an elegant, concise, and powerful language that is useful for tasks large and small. Python has quickly become a popular language for getting things done efficiently in many in all domains: scripting, systems programming, research tools, and web development. This course will provide an introduction to this modern high-level language using hands-on experience through programming assignments and a collaborative final application development project.

Not Offered Every Year  
Prerequisite: CIS 1200  
0-0.5 Course Units

**CIS 1903 Go Programming**

Go is an open source programming language created by Google designed for speed, efficiency and infrastructure. While Go is particularly proficient at concurrent systems programming, it has a variety of uses and has been gaining popularity in a variety of fields, including graphics, mobile applications and machine learning. Go is simple, fast and is continuing to rapidly grow in industry. In this course, we will cover what makes Go so unique and apply it to practical, real world situations. Topics covered will include concurrency and parallelism, goroutines and channels, web scraping, and other popular industry Go applications.

Not Offered Every Year  
Prerequisite: CIS 1100  
0-0.5 Course Units

**CIS 1904 Introduction to Haskell Programming**

Haskell is a high-level, purely functional programming language with a strong static type system and elegant mathematical underpinnings. It is being increasingly used in industry by organizations such as Facebook, AT&T, and NASA, along with several financial firms. We will explore the joys of functional programming, using Haskell as a vehicle. The aim of the course will be to allow you to use Haskell to easily and conveniently write practical programs. Evaluation will be based on regular homework assignments and class participation.

Not Offered Every Year  
Prerequisite: CIS 1200  
0-0.5 Course Units

**CIS 1905 Rust Programming**

Rust is a new, practical, community-developed systems programming language that "runs blazingly fast, prevents almost all crashes, and eliminates data ra (rust-lang.org). Rust derives from a rich history of languages to create a multi-paradigm (imperative/functional), low-level language that focuses on high-performance, zero-cost safety guarantee in concurrent programs. It has begun to gain traction in industry, showing a recognized need for a new low-level systems language. In this course, we will cover what makes Rust so unique and apply it to practical systems programming problems. Topics covered will include traits and generics; memory safety (move semantics, borrowing, and lifetimes); Rust's rich macro system; closures; and concurrency. Evaluation is based on regular homework assignments as well as a final project and class participation. Prerequisite: CIS 1200 Recommended additional prerequisite: CIS 2400 or exposure to C or C++

Not Offered Every Year  
Prerequisite: CIS 1200  
0-0.5 Course Units

**CIS 1911 Using and Understanding Unix and Linux**

Unix, in its many forms, runs much of the world's computer infrastructure, from cable modems and cell phones to the giant clusters that power Google and Amazon. This half-credit course provides a thorough introduction to Unix and Linux. Topics will range from critical basic skills such as examining and editing files, compiling programs and writing shell scripts, to higher level topics such as the architecture of Unix and its programming model. The material learned is applicable to many classes, including CIS 2400, CIS 3310, CIS 3410, CIS 3710, and CIS 3800.

Not Offered Every Year  
Prerequisite: CIS 1100  
0-0.5 Course Units

**CIS 1912 DevOps**

DevOps is the breaking down of the wall between Developers and Operations to allow more frequent and reliable feature deployments. Through a variety of automation-focused techniques, DevOps has the power to radically improve and streamline processes that in the past were manual and susceptible to human error. In this course we will take a practical, hands-on look at DevOps and dive into some of the main tools of DevOps: automated testing, containerization, reproducibility, continuous integration, and continuous deployment. Throughout the semester we build toward an end-to-end pipeline that takes a webserver, packages it, and then deploys it to the cloud in a reliable and quickly-reproducible manner utilizing industry-leading technologies like Kubernetes and Docker. Evaluation is based on homework assignments and a final group project.

Not Offered Every Year  
0-0.5 Course Units

**CIS 1921 Solving Hard Problems in Practice**

What does Sudoku have in common with debugging, scheduling exams, and routing shipments? All of these problems are provably hard – no one has a fast algorithm to solve them. But in reality, people are quickly solving these problems on a huge scale with clever systems and heuristics! In this course, we'll explore how researchers and organizations like Microsoft, Google, and NASA are solving these hard problems, and we'll get to use some of the tools they've built!

Not Offered Every Year  
Prerequisite: CIS 1210  
0-0.5 Course Units

**CIS 1951 iOS Programming**

This project-oriented course is centered around application development on current iOS mobile platforms. The first half of the course will involve fundamentals of mobile app development, where students learn about mobile app lifecycles, event-based programming, efficient resource management, and how to interact with the range of sensors available on modern mobile devices. In the second half of the course, students work in teams to conceptualize and develop a significant mobile application. Creativity and originality are highly encouraged! Prerequisite: CIS 1200 or previous programming experience.

Not Offered Every Year

Prerequisite: CIS 1200

0-0.5 Course Units

**CIS 1952 Android Programming**

This project-oriented course is centered around application development on current Android mobile platforms. The first half of the course will involve fundamentals of mobile app development, where students learn about mobile app lifecycles, event-based programming, efficient resource management, and how to interact with the range of sensors available on modern mobile devices. In the second half of the course, students work in teams to conceptualize and develop a significant mobile application. Creativity and originality are highly encouraged! Prerequisite: CIS 1200 or previous programming experience.

Not Offered Every Year

0-0.5 Course Units

**CIS 1961 Ruby on Rails Web Development**

This course will teach the fundamentals of developing web applications using Ruby on Rails, a rapid-development web framework developed by Basecamp, and adopted by companies like Airbnb, GitHub, Bloomberg, CrunchBase, and Shopify. The first part of the course will focus on Ruby, the language that powers Rails. Along the way, students will also pick up essential skills such as git, bash, HTML and CSS. The second part will focus on Rails, the web framework and will include all topics required to develop and deploy production-ready modern web applications with Rails. Throughout the course, students will be working on a web application project of their own choosing. Upon completion of the course, this application will be deployed and made accessible to the public.

Not Offered Every Year

Prerequisite: CIS 1200

0-0.5 Course Units

**CIS 1962 JavaScript Programming**

This course provides an introduction to modern web development frameworks, techniques, and practices used to deliver robust client side applications on the web. The emphasis will be on developing JavaScript programs that run in the browser. Topics covered include the JavaScript language, web browser internals, the Document Object Model (DOM), HTML5, client-side app architecture and compile-to-JS languages like (CoffeeScript, TypeScript, etc.). This course is most useful for students who have some programming and web development experience and want to develop moderate JavaScript skills to be able to build complex, interactive applications in the browser.

Not Offered Every Year

0-0.5 Course Units

**CIS 1990 Special Topics**

This course will be used for 'pilot versions' of new CIS courses of this type that the department is planning to offer. A given course will be offered as CIS 1990 at most twice; after this, it will be assigned a permanent course number.

0-0.5 Course Units

**CIS 2330 Introduction to Blockchain**

Blockchain or Distributed Ledger Technology (DLT) provides a decentralized method of information sharing between parties that do not trust each other. Instead the trust is in the underlying cryptographic algorithms. This practical introductory course provides experience with the fundamentals of cryptography (codes and ciphers, symmetric and asymmetric encryption, public and private keys, hashes, and zero knowledge proofs) - as it is applied to implementing a blockchain solution. This course covers the basics of a distributed ledger, how it is built, used, and secured. Methods of ensuring consensus - from proof-of-work to proof-of-stake will be explored and analyzed. Students will have both written and practical assignments to build and deploy components of a blockchain solution.

Spring

Prerequisite: CIS 1200

1 Course Unit

**CIS 2400 Introduction to Computer Systems**

You know how to program, but do you know how computers really work? How do millions of transistors come together to form a complete computing system? This bottom-up course begins with transistors and simple computer hardware structures, continues with low-level programming using primitive machine instructions, and finishes with an introduction to the C programming language. This course is a broad introduction to all aspects of computer systems architecture and serves as the foundation for subsequent computer systems courses, such as Computer Organization and Design (CIS 4710), Computer Operating Systems (CIS 3800), and Compilers and Interpreters (CIS 3410).

Fall or Spring

Prerequisite: CIS 1200

1 Course Unit

**CIS 2450 Big Data Analytics**

In the new era of big data, we are increasingly faced with the challenges of processing vast volumes of data. Given the limits of individual machines (compute power, memory, bandwidth), increasingly the solution is to process the data in parallel on many machines. This course focuses on the fundamentals of scaling computation to handle common data analytics tasks. You will learn about basic tasks in collecting, wrangling, and structuring data; programming models for performing certain kinds of computation in a scalable way across many compute nodes; common approaches to converting algorithms to such programming models; standard toolkits for data analysis consisting of a wide variety of primitives; and popular distributed frameworks for analytics tasks such as filtering, graph analysis, clustering, and classification. Recommended: broad familiarity with probability and statistics, as well as programming in Python. Additional background in statistics, data analysis (e.g., in Matlab or R), and machine learning is helpful (e.g., ESE 5420).

Fall or Spring

Mutually Exclusive: CIS 5450

Prerequisite: CIS 1200

1 Course Unit

**CIS 2610 Discrete Probability, Stochastic Processes, and Statistical Inference**

The purpose of this course is to provide a 1 CU educational experience which tightly integrates the theory and applications of discrete probability, discrete stochastic processes, and discrete statistical inference in the study of computer science. The intended audience for this class is both those students who are CS majors as well as those intending to be CS majors. Specifically, it will be assumed that the students will know: Set Theory, Mathematical Induction, Number Theory, Functions, Equivalence Relations, Partial-Order Relations, Combinatorics, and Graph Theory at the level currently covered in CIS 1600. This course could be taken immediately following CIS 1600. Computation and Programming will play an essential role in this course. The students will be expected to use the Maple programming environment in homework exercises which will include: numerical and symbolic computations, simulations, and graphical displays.

Fall

Prerequisite: CIS 1600

1 Course Unit

**CIS 2620 Automata, Computability, and Complexity**

This course explores questions fundamental to computer science such as which problems cannot be solved by computers, can we formalize computing as a mathematical concept without relying upon the specifics of programming languages and computing platforms, and which problems can be solved efficiently. The topics include finite automata and regular languages, context-free grammars and pushdown automata, Turing machines and undecidability, tractability and NP-completeness. The course emphasizes rigorous mathematical reasoning as well as connections to practical computing problems such as test processing, parsing, XML query languages, and program verification.

Fall

Prerequisite: CIS 1600

1 Course Unit

**CIS 3200 Introduction to Algorithms**

How do you optimally encode a text file? How do you find shortest paths in a map? How do you design a communication network? How do you route data in a network? What are the limits of efficient computation? This course gives a comprehensive introduction to design and analysis of algorithms, and answers along the way to these and many other interesting computational questions. You will learn about problem-solving; advanced data structures such as universal hashing and red-black trees; advanced design and analysis techniques such as dynamic programming and amortized analysis; graph algorithms such as minimum spanning trees and network flows; NP-completeness theory; and approximation algorithms.

Fall or Spring

Prerequisite: CIS 1210 AND CIS 2620

1 Course Unit

**CIS 3333 Mathematics of Machine Learning**

Machine learning is the study of algorithms (e.g. gradient descent) that learn functions (e.g. deep networks) from experience (e.g. data). Behind this simple statement is a lot of mathematical scaffolding: statistics for handling data, optimization for understanding learning algorithms, and linear algebra to create expressive models. This course provides the background to be able to understand mathematical concepts commonly used in machine learning. Topics include continuous probability, parametric distributions, and concentration inequalities from statistics; inner product spaces, functional analysis and Hilbert spaces from linear algebra; and multivariate calculus, Taylor's theorem, and convexity from optimization.

Fall or Spring

Prerequisite: CIS 1600 AND STAT 4300 AND MATH 2400

1 Course Unit

**CIS 3340 Advanced Topics in Algorithms**

Can you check if two large documents are identical by examining a small number of bits? Can you verify that a program has correctly computed a function without ever computing the function? Can students compute the average score on an exam without ever revealing their scores to each other? Can you be convinced of the correctness of an assertion without ever seeing the proof? The answer to all these questions is in the affirmative provided we allow the use of randomization. Over the past few decades, randomization has emerged as a powerful resource in algorithm design. This course would focus on powerful general techniques for designing randomized algorithms as well as specific representative applications in various domains, including approximation algorithms, cryptography and number theory, data structure design, online algorithms, and parallel and distributed computation.

Not Offered Every Year

Prerequisite: CIS 3200

1 Course Unit

**CIS 3410 Compilers and Interpreters**

You know how to program, but do you know how to implement a programming language? In CIS 3410 you'll learn how to build a compiler. Topics covered include: lexical analysis, grammars and parsing, intermediate representations, syntax-directed translation, code generation, type checking, simple dataflow and control-flow analyses, and optimizations. Along the way, we study objects and inheritance, first-class functions (closures), data representation and runtime-support issues such as garbage collection. This is a challenging, implementation-oriented course in which students build a full compiler from a simple, typed object-oriented language to fully operational x86 assembly. The course projects are implemented using OCaml, but no knowledge of OCaml is assumed. Prerequisite: Two semesters of programming courses, e.g., CIS 1200, CIS 1210, CIS 2400.

Not Offered Every Year

1 Course Unit

**CIS 3500 Software Design/Engineering**

You know how to write a "program". But how do you create a software "product" as part of a team, with customers that have expectations of functionality and quality? This course introduces students to various tools (source control, automated build systems, programming environments, test automation, etc.) and processes (design, implementation, testing, and maintenance) that are used by professionals in the field of software engineering. Topics will include: software development lifecycle; agile and test-driven development; source control and continuous integration; requirements analysis; object-oriented design and testability; mobile and/or web application development; software testing; refactoring ; and software quality metrics.

Not Offered Every Year

Prerequisite: CIS 1210

1 Course Unit

**CIS 3900 Robotics: Planning Perception**

This introductory course will present basic principles of robotics with an emphasis to computer science aspects. Algorithms for planning and perception will be studied and implemented on actual robots. While planning is a fundamental problem in artificial intelligence and decision making, robot planning refers to finding a path from A to B in the presence of obstacles and by complying with the kinematic constraints of the robot. Perception involves the estimation of the robots motion and path as well as the shape of the environment from sensors. In this course, algorithms will be implemented in Python on mobile platforms on ground and in the air. No prior experience with Python is needed but we require knowledge of data structures, linear algebra, and basic probability.

Not Offered Every Year

Prerequisite: CIS 1210 AND MATH 2400

1 Course Unit

**CIS 3980 Quantum Computer and Information Science**

The purpose of this course is to introduce undergraduate students in computer computer science and engineering to quantum computers (QC) and quantum information science (QIS). This course is meant primarily for juniors and seniors in Computer Science. No prior knowledge of quantum mechanics (QM) is assumed. Enrollment is by permission of the instructor.

Fall or Spring

Prerequisite: (PHYS 0151 OR PHYS 0171) AND MATH 2400 AND MATH 3120 AND MATH 3140 AND CIS 1600 AND CIS 2620

1 Course Unit

**CIS 3990 Special Topics**

Visit the CIS department website for descriptions of available Special Topics classes.

Not Offered Every Year

0.5-1 Course Unit

**CIS 4000 Senior Project**

Design and implementation of a significant piece of work: software, hardware or theory. In addition, emphasis on technical writing and oral communication skills. Students must have an abstract of their Senior Project, which is approved and signed by a Project Adviser, at the end of the second week of Fall classes. The project continues during two semesters; students must enroll in CIS 401 during the second semester. At the end of the first semester, students are required to submit an intermediate report and give a class presentation describing their project and progress. Grades are based on technical writing skills (as per submitted report), oral presentation skills (as per class presentation) and progress on the project. These are evaluated by the Project Adviser and the Course Instructor. Senior standing or permission of instructor.

Fall

1 Course Unit

**CIS 4010 Senior Project**

Continuation of CIS 4000. Design and implementation of a significant piece of work: software, hardware or theory. Students are required to submit a final written report and give a final presentation and demonstration of their project. Grades are based on the report, the presentation and the satisfactory completion of the project. These are evaluated by the Project Adviser and the Course Instructor. Senior standing or permission of instructor.

Spring

Prerequisite: CIS 4000

1 Course Unit

**CIS 4100 CIS Senior Thesis**

The goal of a Senior Thesis project is to complete a major research project under the supervision of a faculty member. The duration of the project is two semesters. To enroll in CIS 4100, students must develop an abstract of the proposed work, and a member of the CIS graduate group must certify that the work is suitable and agree to supervise the project; a second member must agree to serve as a reader. At the end of the first semester, students must submit an intermediate report; if the supervisor and reader accept it, they can enroll in CIS 4110. At the end of the second semester, students must describe their results in a written thesis and must present them publicly, either in a talk at Penn or in a presentation at a conference or workshop. Grades are based on the quality of the research itself (which should ideally be published or at least of publishable quality), as well as on the quality of the thesis and the oral presentation. The latter are evaluated jointly by the supervisor and the reader. The Senior Thesis program is selective, and students are generally expected to have a GPA is in the top 10-20% to qualify. Senior Theses are expected to integrate the knowledge and skills from earlier coursework; because of this, students are not allowed to enroll in CIS 4100 before their sixth semester.

1 Course Unit

**CIS 4110 CIS Senior Thesis**

The goal of a Senior Thesis project is to complete a major research project under the supervision of a faculty member. The duration of the project is two semesters. To enroll in CIS 4100, students must develop an abstract of the proposed work, and a member of the CIS graduate group must certify that the work is suitable and agree to supervise the project; a second member must agree to serve as a reader. At the end of the first semester, students must submit an intermediate report; if the supervisor and reader accept it, they can enroll in CIS 4110. At the end of the second semester, students must describe their results in a written thesis and must present them publicly, either in a talk at Penn or in a presentation at a conference or workshop. Grades are based on the quality of the research itself (which should ideally be published or at least of publishable quality), as well as on the quality of the thesis and the oral presentation. The latter are evaluated jointly by the supervisor and the reader. The Senior Thesis program is selective, and students are generally expected to have a GPA is in the top 10-20% to qualify.

1 Course Unit

**CIS 4120 Introduction to Human Computer Interaction**

In this course, you will learn the essentials of human-computer interaction (HCI). Over the course of a semester, you will learn how to design interactive systems that satisfy and delight users by undertaking the human-centered design process, from ideation to prototyping, implementation, and assessment with human users. You will learn key tools in the HCI toolkit, including need-finding, user studies, visual design, cognitive models, demo'ing, ethical considerations, and writing about your designs. This course also provides a primer on several areas of emerging technology in HCI, such as human-AI interaction and education technology. We will also cover ethics in HCI, including topics like inclusive design and dark patterns. To hone your craft as an HCI practitioner, during this course you will undertake a group project to design an innovative user interface. The final submission will include a working interactive prototype, demonstrations of the interface at a public departmental design showcase, and a written reflection on your design findings. Prerequisite: prior programming experience

Fall

Prerequisite: Prior programming experience

1 Course Unit

**CIS 4190 Applied Machine Learning**

Machine learning has been essential to the success of many recent technologies, including autonomous vehicles, search engines, genomics, automated medical diagnosis, image recognition, and social network analysis, among many others. This course will introduce the fundamental concepts and algorithms that enable computers to learn from experience, with an emphasis on their practical application to real problems. This course will introduce supervised learning (decision trees, logistic regression, support vector machines, Bayesian methods, neural networks and deep learning), unsupervised learning (clustering, dimensionality reduction), and reinforcement learning. Additionally, the course will discuss evaluation methodology and recent applications of machine learning, including large scale learning for big data and network analysis.

Fall or Spring

Mutually Exclusive: CIS 5190

Prerequisite: CIS 1210

1 Course Unit

**CIS 4210 Artificial Intelligence**

This course investigates algorithms to implement resource-limited knowledge-based agents which sense and act in the world. Topics include, search, machine learning, probabilistic reasoning, natural language processing, knowledge representation and logic. After a brief introduction to the language, programming assignments will be in Python.

Fall

Mutually Exclusive: CIS 5210

Prerequisite: CIS 1210

1 Course Unit

**CIS 4230 Ethical Algorithm Design**

This class introduces aspiring data science technologists to the spectrum of ethical concerns, focusing on social norms like fairness, transparency and privacy. It introduces technical approaches to a number of these problems, including by hands-on examination of the tradeoffs in fairness and accuracy in predictive technology, introduction to differential privacy, and overview of evaluation conventions for predictive technology. It also provides guidelines for examining system training data for bias, representation (of race, gender and other characteristics) and ecological validity. Equipped with this knowledge, students will learn how to conduct informed analysis of the usefulness of predictive systems; they will audit for ethical concerns papers from the contemporary top artificial intelligence venues and the ongoing senior design projects.

Fall or Spring

Mutually Exclusive: CIS 5230

Prerequisite: CIS 1210

1 Course Unit

**CIS 4300 Natural Language Processing**

This course provides an overview of the field of natural language processing. The goal of the field is to build technologies that will allow machines to understand human languages. Applications include machine translation, automatic summarization, question answering systems, and dialog systems. NLP is used in technologies like Amazon Alexa and Google Translate.

Fall

Mutually Exclusive: CIS 5300

Prerequisite: CIS 1210 AND CIS 1600

1 Course Unit

**CIS 4360 Introduction to Computational Biology & Biological Modeling**

The goal of this course is to develop a deeper understanding of techniques and concepts used in Computational Biology. Both theoretical and practical aspects of a range of methods will be covered. Theoretical aspects will include statistical analysis, modeling, and algorithm design. This course cannot provide a comprehensive survey of the field but focuses on a select core set of topics and data types. We will discuss the genome browser, alignment algorithms, classical and non-parametric statistics, pathway analysis, dimensionality reduction, GWAS, multiple testing and machine learning, with primary focus on biomedical data. UNIX, R and Python will be utilized to learn to execute big data analysis pipelines, including RNA-Seq and DNA-Seq. UNIX and R will be taught from first principles but programming experience in Python is expected. Students without prior experience with Python should consider taking PHYS 1100 before taking this class. You will be provided with a computational (cloud based) platform on which to do all programming and assignments.

Fall

Also Offered As: BIOL 4536

Mutually Exclusive: BIOL 5535

Prerequisite: MATH 1400 AND (BIOL 2510 OR BIOL 5510)

1 Course Unit

**CIS 4410 Embedded Software for Life-Critical Applications**

The goal of this course is to give students greater design and implementation experience in embedded software development and to teach them how to model, design, verify, and validate safety critical systems in a principled manner. Students will learn the principles, methods, and techniques for building life-critical embedded systems, ranging from requirements and models to design, analysis, optimization, implementation, and validation. Topics will include modeling and analysis methods and tools, real-time programming paradigms and languages, distributed real-time systems, global time, time-triggered communications, assurance case, software architecture, evidence-based certification, testing, verification, and validation. The course will include a series of projects that implements life-critical embedded systems (e.g., pacemaker, infusion pumps, closed-loop medical devices).

Fall

Prerequisite: CIS 2400

1 Course Unit

**CIS 4480 Operating Systems Design and Implementation**

The purpose of this course is to teach the design and implementation of operating systems and operating systems concepts that appear in other advanced systems. The course divides into three major sections. The first part of the course discusses concurrency: how to manage multiple tasks that execute at the same time and share resources. Topics in this section include processes and threads, context switching, synchronization, scheduling, and deadlock. The second part of the course addresses the problem of memory management; it will cover topics such as linking, dynamic memory allocation, dynamic address translation, virtual memory, and demand paging. The third major part of the course concerns file systems, including topics such as storage devices, disk management and scheduling, directories, protection, and crash recovery. After these three major topics, the class will conclude with specialized topics such as virtual machines and case studies of different operating systems (e.g. Android, Windows, Linux, etc.).

Fall

Mutually Exclusive: CIS 5480

Prerequisite: CIS 2400

1 Course Unit

**CIS 4500 Database and Information Systems**

This course provides an introduction to the broad field of database and information systems, covering a variety of topics relating to structured data, ranging from data modeling to logical foundations and popular languages, to system implementations. We will study the theory of relational and XML data design; the basics of query languages; efficient storage of data, execution of queries and query optimization; transactions and updates; web-database development; and "big data" and NoSQL systems. The course assumes mathematical and programming experience equivalent to CIS 1600 and CIS 1210.

Fall or Spring

Mutually Exclusive: CIS 5500

Prerequisite: CIS 1210 AND CIS 1600

1 Course Unit

**CIS 4510 Computer and Network Security**

This is an introduction to topics in the security of computer systems and communication on networks of computers. The course covers four major areas: fundamentals of cryptography, security for communication protocols, security for operating systems and mobile programs, and security for electronic commerce. Sample specific topics include: passwords and offline attacks, DES, RSA, DSA, SHA, SSL, CBC, IPSec, SET, DDoS attacks, biometric authentication, PKI, smart cards, S/MIME, privacy on the Web, viruses, security models, wireless security, and sandboxing. Students will be expected to display knowledge of both theory and practice through written examinations and programming assignments.

Fall or Spring

Mutually Exclusive: CIS 5510

Prerequisite: CIS 1600 AND CIS 2400

1 Course Unit

**CIS 4550 Internet and Web Systems**

This course focuses on the challenges encountered in building Internet and web systems: scalability, interoperability (of data and code), security and fault tolerance, consistency models, and location of resources, services, and data. We will examine how XML standards enable information exchange; how web services support cross-platform interoperability (and what their limitations are); how to build high-performance application servers; how "cloud computing" services work; how to perform Akamai-like content distribution; and how to provide transaction support in distributed environments. We will study techniques for locating machines, resources, and data (including directory systems, information retrieval indexing, ranking, and web search); and we will investigate how different architectures support scalability (and the issues they face). We will also examine ideas that have been proposed for tomorrow's Web, and we will see some of the challenges, research directions, and potential pitfalls. An important goal of the course is not simply to discuss issues and solutions, but to provide hands-on experience with a substantial implementation project. This semester's project will be a peer-to-peer implementation of a Google-style search engine, including distributed, scalable crawling; indexing with ranking; and even PageRank. As a side-effect of the material of this course you will learn about some aspects of large-scale software development assimilating large APIs. Prerequisite: Familiarity with threads and concurrency, strong Java programming skills.

Fall or Spring

Mutually Exclusive: CIS 5550

1 Course Unit

**CIS 4600 Interactive Computer Graphics**

This course focuses on programming the essential mathematical and geometric concepts underlying modern computer graphics. Using 3D interactive implementations, it covers fundamental topics such as mesh data structures, transformation sequences, rendering algorithms, and curve interpolation for animation. Students are also introduced to two programming languages widely used in the computer graphics industry: C++ and GLSL. The curriculum is heavily project-based, and culminates in a group project focused on building an interactive first-person world exploration application using the various real-time interaction and rendering algorithms learned throughout the semester. The prerequisite for this course is CIS 1210. We also recommend CIS 2400.

Fall or Spring

Prerequisite: CIS 1210

1 Course Unit

**CIS 4610 Advanced Rendering**

This course is designed to provide a comprehensive overview to computer graphics techniques in 3D modeling, image synthesis, and rendering. Topics cover: geometric transformations, geometric algorithms, software systems, 3D object models (surface, volume and implicit), visible surface algorithms, image synthesis, shading, mapping, ray tracing, radiosity, global illumination, sampling, anti-aliasing, Monte Carlo path tracing, and photon mapping. Prerequisite: A working knowledge of C++ programming is required (one year programming experience in general). Knowledge of vector geometry is useful.

Fall or Spring

Mutually Exclusive: CIS 5610

1 Course Unit

**CIS 4620 Computer Animation**

This course covers core subject matter common to the fields of robotics, character animation and embodied intelligent agents. The intent of the course is to provide the student with a solid technical foundation for developing, animating and controlling articulated systems used in interactive computer game virtual reality simulations and high-end animation applications. The course balances theory with practice by "looking under the hood" of current animation systems and authoring tools and exams the technologies and techniques used from both a computer science and engineering perspective. Topics covered include: geometric coordinate systems and transformations; quaternions; parametric curves and surfaces; forward and inverse kinematics; dynamic systems and control; computer simulation; keyframe, motion capture and procedural animation; behavior-based animation and control; facial animation; smart characters and intelligent agents. Prerequisite: Previous exposure to major concepts in linear algebra (i.e. vector matrix math), curves and surfaces, dynamical systems (e.g. 2nd order mass-spring-damper systems) and 3D computer graphics has also been assumed in the preparation of the course materials.

Fall or Spring

Mutually Exclusive: CIS 5620

1 Course Unit

**CIS 4670 Scientific Computing**

This course will focus on numerical algorithms and scientific computing techniques that are practical and efficient for a number of canonical science and engineering applications. Built on top of classical theories in multi-variable calculus and linear algebra (as a prerequisite), the lectures in this course will strongly focus on explaining numerical methods for applying these mathematical theories to practical engineering problems. Students will be expected to implement solutions and software tools using MATLAB/C++, practice state-of-the-art parallel computing paradigms, and learn scientific visualization techniques using modern software packages. Prerequisites: MATH 240; knowledge of C++, Python or MATLAB

Mutually Exclusive: CIS 5670

Prerequisite: MATH 2400

1 Course Unit

**CIS 4710 Computer Organization and Design**

This is the second computer organization course and focuses on computer hardware design. Topics covered are: (1) basic digital system design including finite state machines, (2) instruction set design and simple RISC assembly programming, (3) quantitative evaluation of computer performance, (4) circuits for integer and floating-point arithmetic, (5) datapath and control, (6) micro-programming, (7) pipelining, (8) storage hierarchy and virtual memory, (9) input/output, (10) different forms of parallelism including instruction level parallelism, data-level parallelism using both vectors and message-passing multi-processors, and thread-level parallelism using shared memory multiprocessors. Basic cache coherence and synchronization. Prerequisite: Knowledge of at least one programming language.

Spring

Mutually Exclusive: CIS 5710

Prerequisite: CIS 2400

1 Course Unit

**CIS 4810 Computer Vision & Computational Photography**

This is an introductory course to Computer Vision and Computational Photography. This course will explore three topics: 1) image morphing, 2) image matching and stitching, and 3) image recognition. This course is intended to provide a hands-on experience with interesting things to do on images/videos. The world is becoming image-centric. Cameras are now found everywhere, in our cell phones, automobiles, even in medical surgery tools. Computer vision technology has led to latest innovations in areas such as Hollywood movie production, medical diagnosis, biometrics, and digital library. This course is suited for students from all Engineering backgrounds, who have the basic knowledge of linear algebra and programming, and a lot of imagination.

Fall

Mutually Exclusive: CIS 5810

Prerequisite: CIS 1200 AND CIS 1600

1 Course Unit

**CIS 4970 DMD Senior Project**

The goal of this course is to provide an opportunity for seniors to define, design and execute a project of your own choosing that demonstrates the technical skills and abilities that you have acquired during your 4 years as undergraduates. Evaluation is based on selecting an interesting topic, completing appropriate research on the state of the art in that area, communicating your objectives in writing and in presentations, accurately estimating what resources will be required to complete your chosen task, coding necessary functionality, and executing your plan. Senior Standing or Permission of the instructor.

Fall or Spring

1 Course Unit



**CIS 4980 Senior Capstone Project**

The Senior Capstone Project is required for all BAS degree students, in lieu of the senior design course. The Capstone Project provides an opportunity for the student to apply the theoretical ideas and tools learned from other courses. The project is usually applied, rather than theoretical, exercise, and should focus on a real-world problem related to the career goals of the student. The one-semester project may be completed in either the fall or spring term of the senior year, and must be done under the supervision of a sponsoring faculty member. To register for this course, the student must submit a detailed proposal, signed by the supervising professor and the student's faculty advisor, two weeks prior to the start of the term.

Fall or Spring  
1 Course Unit

**CIS 5000 Software Foundations**

This course introduces basic concepts and techniques in the foundational study of programming languages. The central theme is the view of programs and programming languages as mathematical objects for which precise claims may be made and proved. Particular topics include operational techniques for formal definition of language features, type systems and type safety properties, polymorphism, constructive logic, and the Coq proof assistant. This course is appropriate as an upper-level undergraduate CIS elective. Undergraduates who have satisfied the prerequisites are welcome to enroll. There are no formal prerequisites, but students should know how to program at the level of a couple of undergrad courses and have developed some "mathematical maturity" (through two or more undergraduate math courses, including at least one involving rigorous proofs); a course in functional programming or compilers may also be helpful.

Fall or Spring  
1 Course Unit

**CIS 5010 Computer Architecture**

This course is an introductory graduate course on computer architecture with an emphasis on a quantitative approach to cost/performance design tradeoffs. The course covers the fundamentals of classical and modern uniprocessor design: performance and cost issues, instruction sets, pipelining, superscalar, out-of-order, and speculative execution mechanisms, caches, physical memory, virtual memory, and I/O. Other topics include: static scheduling, VLIW and EPIC, software speculation, long (SIMD) and short (multimedia) vector execution, multithreading, and an introduction to shared memory multiprocessors. Knowledge of computer organization and basic programming skills.

Fall or Spring  
1 Course Unit

**CIS 5020 Analysis of Algorithms**

An investigation of paradigms for design and analysis of algorithms. The course will include dynamic programming, flows and combinatorial optimization algorithms, linear programming, randomization and a brief introduction to intractability and approximation algorithms. The course will include other advanced topics, time permitting. Prerequisite: Data Structures and Algorithms at the undergraduate level.

Fall or Spring  
1 Course Unit

**CIS 5050 Software Systems**

This course provides an introduction to fundamental concepts of distributed systems, and the design principles for building large scale computational systems. Topics covered include communication, concurrency, programming paradigms, naming, managing shared state, caching, synchronization, reaching agreement, fault tolerance, security, middleware, and distributed applications. This course is appropriate as an upper-level undergraduate CIS elective. Prerequisite: Undergraduate-level knowledge of Operating Systems and Networking, programming experience. Prerequisite: Undergraduate-level knowledge of Operating Systems and Networking.

Fall or Spring  
1 Course Unit

**CIS 5110 Theory of Computation**

Review of regular and context-free languages and machine models. Turing machines and RAM models, Decidability, Halting problem, Reductions, Recursively enumerable sets, Universal TMs, Church/Turing thesis. Time and space complexity, hierarchy theorems, the complexity classes P, NP, PSPACE, L, NL, and co-NL. Reductions revisited, Cook-Levin Theorem, completeness, NL = co-NL. Advanced topics as time permits: Circuit complexity and parallel computation, randomized complexity, approximability, interaction and cryptography. Discrete Mathematics, Automata theory or Algorithms at the undergraduate level.

Fall or Spring  
1 Course Unit

**CIS 5120 Introduction to Human Computer Interaction**

In this course, you will learn the essentials of human-computer interaction (HCI). Over the course of a semester, you will learn how to design interactive systems that satisfy and delight users by undertaking the human-centered design process, from ideation to prototyping, implementation, and assessment with human users. You will learn key tools in the HCI toolkit, including need-finding, user studies, visual design, cognitive models, demo'ing, ethical considerations, and writing about your designs. This course also provides a primer on several areas of emerging technology in HCI, such as human-AI interaction and education technology. To hone your craft as an HCI practitioner, during this course you will undertake a group project to design an innovative user interface. The final submission will include a working interactive prototype, demonstrations of the interface at a public departmental design showcase, and a written reflection on your design findings.

Prerequisite: prior programming experience  
Fall

Mutually Exclusive: CIS 4120  
Prerequisite: Prior programming experience  
1 Course Unit

**CIS 5150 Fundamentals of Linear Algebra and Optimization**

This course provides firm foundations in linear algebra and basic optimization techniques. Emphasis is placed on teaching methods and tools that are widely used in various areas of computer science. Both theoretical and algorithmic aspects will be discussed.

Fall or Spring  
1 Course Unit

**CIS 5180 Topics in Logic: Finite Model Theory and Descriptive Complexity**

This course will examine the expressive power of various logical languages over the class of finite structures. The course begins with an exposition of some of the fundamental theorems about the behavior of first-order logic in the context of finite structures, in particular, the Ehrenfeucht-Fraïssé Theorem and the Trakhtenbrot Theorem. The first of these results is used to show limitations on the expressive power of first-order logic over finite structures while the second result demonstrates that the problem of reasoning about finite structures using first-order logic is surprisingly complex. The course then proceeds to consider various extensions of first-order logic including fixed-point operators, generalized quantifiers, infinitary languages, and higher-order languages. The expressive power of these extensions will be studied in detail and will be connected to various problems in the theory of computational complexity. This last motif, namely the relation between descriptive and computational complexity, will be one of the main themes of the course.

Fall or Spring

1 Course Unit

**CIS 5190 Applied Machine Learning**

Machine learning has been essential to the success of many recent technologies, including autonomous vehicles, search engines, genomics, automated medical diagnosis, image recognition, and social network analysis, among many others. This course will introduce the fundamental concepts and algorithms that enable computers to learn from experience, with an emphasis on their practical application to real problems. This course will introduce supervised learning (decision trees, logistic regression, support vector machines, Bayesian methods, neural networks and deep learning), unsupervised learning (clustering, dimensionality reduction), and reinforcement learning. Additionally, the course will discuss evaluation methodology and recent applications of machine learning, including large scale learning for big data and network analysis. Preparation for this course would include taking CIS 1210 or equivalent knowledge.

Fall or Spring

Mutually Exclusive: CIS 4190

1 Course Unit

**CIS 5200 Machine Learning**

This course covers the foundations of statistical machine learning. The focus is on probabilistic and statistical methods for prediction and clustering in high dimensions. Topics covered include linear and logistic regression, SVMs, PCA and dimensionality reduction, EM and HMMs, and deep learning. Elementary probability, calculus, and linear algebra. Basic programming experience.

Fall

1 Course Unit

**CIS 5210 Artificial Intelligence**

This course investigates algorithms to implement resource-limited knowledge-based agents which sense and act in the world. Topics include, search, machine learning, probabilistic reasoning, natural language processing, knowledge representation and logic. After a brief introduction to the language, programming assignments will be in Python.

Fall or Spring

Mutually Exclusive: CIS 4210

1 Course Unit

**CIS 5220 Deep Learning for Data Science**

Deep learning techniques now touch on data systems of all varieties. Sometimes, deep learning is a product; sometimes, deep learning optimizes a pipeline; sometimes, deep learning provides critical insights; sometimes, deep learning sheds light on neuroscience or vice versa. The purpose of this course is to deconstruct the hype by teaching deep learning theories, models, skills, and applications that are useful for applications.

Fall

1 Course Unit

**CIS 5230 Ethical Algorithm Design**

This class introduces aspiring data science technologists to the spectrum of ethical concerns, focusing on social norms like fairness, transparency and privacy. It introduces technical approaches to a number of these problems, including by hands-on examination of the tradeoffs in fairness and accuracy in predictive technology, introduction to differential privacy, and overview of evaluation conventions for predictive technology. It also provides guidelines for examining system training data for bias, representation (of race, gender and other characteristics) and ecological validity. Equipped with this knowledge, students will learn how to conduct informed analysis of the usefulness of predictive systems; they will audit for ethical concerns papers from the contemporary top artificial intelligence venues and the ongoing senior design projects. Preparation for this course would include taking CIS 1210 or equivalent knowledge.

Fall or Spring

Mutually Exclusive: CIS 4230

1 Course Unit

**CIS 5260 Machine Translation**

Google translate can instantly translate between any pair of over fifty human languages (for instance, from French to English). How does it do that? Why does it make the errors that it does? And how can you build something better? Modern translation systems like Google Translate and Bing Translator learn how to translate by reading millions of words of already translated text, and this course will show you how they work. The course covers a diverse set of fundamental building blocks from linguistics, machine learning, algorithms, data structures, and formal language theory, along with their application to a real and difficult problem in artificial intelligence.

Fall or Spring

1 Course Unit

**CIS 5300 Natural Language Processing**

This course provides an overview of the field of natural language processing. The goal of the field is to build technologies that will allow machines to understand human languages. Applications include machine translation, automatic summarization, question answering systems, and dialog systems. NLP is used in technologies like Amazon Alexa and Google Translate. We highly recommend experience in machine learning to be successful in this course, along with a background in data structures and algorithms.

Fall

Mutually Exclusive: CIS 4300

1 Course Unit

**CIS 5350 Introduction to Bioinformatics**

This course provides overview of bioinformatics and computational biology as applied to biomedical research. A primary objective of the course is to enable students to integrate modern bioinformatics tools into their research activities. Course material is aimed to address biological questions using computational approaches and the analysis of data. A basic primer in programming and operating in a UNIX environment will be presented, and students will also be introduced to Python R, and tools for reproducible research. This course emphasizes direct, hands-on experience with applications to current biological research problems. Areas include DNA sequence alignment, genetic variation and analysis, motif discovery, study design for high-throughput sequencing RNA, and gene expression, single gene and whole-genome analysis, machine learning, and topics in systems biology. The relevant principles underlying methods used for analysis in these areas will be introduced and discussed at a level appropriate for biologists without a background in computer science. The course is not intended for computer science students who want to learn about biologically motivated algorithmic problems; BIOL 4536/BIOL 5536 and BE 5370/CIS 5370/MPHY 6090 are more appropriate. Prerequisites: An advanced undergraduate course such as BIOL 4210 or a graduate course in biology such as BIOL 5210, BIOL 5240, or equivalent, is a prerequisite.

Fall

Also Offered As: BIOM 5350, MTR 5350

Prerequisite: BIOL 4210 OR BIOL 5210 OR BIOL 5240

1 Course Unit

**CIS 5360 Fundamentals of Computational Biology**

Introductory computational biology course designed for both biology students and computer science, engineering students. The course will cover fundamentals of algorithms, statistics, and mathematics as applied to biological problems. In particular, emphasis will be given to biological problem modeling and understanding the algorithms and mathematical procedures at the "pencil and paper" level. That is, practical implementation of the algorithms is not taught but principles of the algorithms are covered using small sized examples. Topics to be covered are: genome annotation and string algorithms, pattern search and statistical learning, molecular evolution and phylogenetics, functional genomics and systems level analysis.

Fall

Also Offered As: BIOL 5536, GCB 5360

Prerequisite: ((BIOL 1101 AND BIOL 1102) OR BIOL 1121) AND STAT 111 AND STAT 112

1 Course Unit

**CIS 5370 Biomedical Image Analysis**

This course covers the fundamentals of advanced quantitative image analysis that apply to all of the major and emerging modalities in biological/biomaterials imaging and in vivo biomedical imaging. While traditional image processing techniques will be discussed to provide context, the emphasis will be on cutting edge aspects of all areas of image analysis (including registration, segmentation, and high-dimensional statistical analysis). Significant coverage of state-of-the-art biomedical research and clinical applications will be incorporated to reinforce the theoretical basis of the analysis methods. Prerequisite: Mathematics through multivariate calculus (MATH 2410), programming experience, as well as some familiarity with linear algebra, basic physics, and statistics.

Fall or Spring

Also Offered As: BE 5370, MPHY 6090

1 Course Unit

**CIS 5400 Principles of Embedded Computation**

This course is focused on principles underlying design and analysis of computational elements that interact with the physical environment. Increasingly, such embedded computers are everywhere, from smart cameras to medical devices to automobiles. While the classical theory of computation focuses on the function that a program computes, to understand embedded computation, we need to focus on the reactive nature of the interaction of a component with its environment via inputs and outputs, the continuous dynamics of the physical world, different ways of communication among components, and requirements concerning safety, timeliness, stability, and performance. Developing tools for approaching design, analysis, and implementation of embedded systems in a principled manner is an active research area. This course will attempt to give students a coherent introduction to this emerging area. This course is appropriate as an upper-level undergraduate CIS elective. This course assumes mathematical maturity, commensurate with either ESE 2100 (Introduction to Dynamical Systems), or CIS 2620 (Introduction to Theory of Computation). It is suitable for students who have an undergraduate degree in computer science, or computer engineering, or electrical engineering. It is also suitable for Penn undergraduates in CIS or CE as an upper-level elective.

Fall

1 Course Unit

**CIS 5410 Embedded Software for Life-Critical Applications**

The goal of this course is to give students greater design and implementation experience in embedded software development and to teach them how to model, design, verify, and validate safety critical systems in a principled manner. Students will learn the principles, methods, and techniques for building life-critical embedded systems, ranging from requirements and models to design, analysis, optimization, implementation, and validation. Topics will include modeling and analysis methods and tools, real-time programming paradigms and languages, distributed real-time systems, global time, time-triggered communications, assurance case, software architecture, evidence-based certification, testing, verification, and validation. The course will include a series of projects that implements life-critical embedded systems (e.g., pacemaker, infusion pumps, closed-loop medical devices). This course assumes experience equivalent to CIS 2400 (Introduction to Computer Systems).

Fall or Spring

1 Course Unit

**CIS 5420 Embedded Systems Programming**

This course explores techniques for writing correct and efficient embedded code. Topics include C/C++ idioms, data abstraction, elementary data structures and algorithms, environment modeling, concurrency, hard real time, and modular program reasoning. C fluency.

Fall or Spring

1 Course Unit

**CIS 5450 Big Data Analytics**

In the new era of big data, we are increasingly faced with the challenges of processing vast volumes of data. Given the limits of individual machines (compute power, memory, bandwidth), increasingly the solution is to process the data in parallel on many machines. This course focuses on the fundamentals of scaling computation to handle common data analytics tasks. You will learn about basic tasks in collecting, wrangling, and structuring data; programming models for performing certain kinds of computation in a scalable way across many compute nodes; common approaches to converting algorithms to such programming models; standard toolkits for data analysis consisting of a wide variety of primitives; and popular distributed frameworks for analytics tasks such as filtering, graph analysis, clustering, and classification. Recommended: broad familiarity with probability and statistics, as well as programming in Python. Additional background in statistics, data analysis (e.g., in Matlab or R), and machine learning is helpful (example : ESE 5420).

Fall or Spring

Mutually Exclusive: CIS 2450

1 Course Unit

**CIS 5470 Software Analysis**

This course covers the theory and practice of software analysis - a body of algorithms and techniques to reason about program behavior with applications to effectively test, debug, and secure large, complex codebases. The course surveys a wide range of applications of software analysis including proving the absence of common programming errors, discovering and preventing security vulnerabilities, systematically testing intricate data structures and libraries, and localizing root causes in complex software failures. Familiarity with programming (CIS 1200), algorithms (CIS 1210), and mathematical foundations (CIS 1600). Specifically: - Assignments involve programming in C/C++ in the LLVM compiler infrastructure. - Lectures and exams presume knowledge of search and graph algorithms, and background in logic and probability.

Fall or Spring

1 Course Unit

**CIS 5480 Operating Systems Design and Implementation**

The purpose of this masters-level course is to teach the design and implementation of operating systems and operating systems concepts that appear in other advanced systems. The course divides into three major sections. The first part of the course discusses concurrency: how to manage multiple tasks that execute at the same time and share resources. Topics in this section include processes and threads, context switching, synchronization, scheduling, and deadlock. The second part of the course addresses the problem of memory management; it will cover topics such as linking, dynamic memory allocation, dynamic address translation, virtual memory, and demand paging. The third major part of the course concerns file systems, including topics such as storage devices, disk management and scheduling, directories, protection, and crash recovery. After these three major topics, the class will conclude with specialized topics such as virtual machines and case studies of different operating systems (e.g. Android, Windows, Linux, etc.).

Fall or Spring

Mutually Exclusive: CIS 4480

1 Course Unit

**CIS 5490 Wireless Communications for Mobile Networks and Internet of Things**

This course covers generations of wireless mobile network standards and systems, basic differences and their evolution, charting the development of mobile telecommunications systems from 3G, to today's state-of-the-art wireless technology 4G LTE, and the next generation wireless technology, 5G. The course projects require knowledge of C/C++. Any undergraduate networking courses are suggested but not required as this course covers necessary networking topics.

Fall or Spring

1 Course Unit

**CIS 5500 Database and Information Systems**

This course provides an introduction to the broad field of database and information systems, covering a variety of topics relating to structured data, ranging from data modeling to logical foundations and popular languages, to system implementations. We will study the theory of relational and XML data design; the basics of query languages; efficient storage of data, execution of queries and query optimization; transactions and updates; web-database development; and "big data" and NoSQL systems. The course assumes mathematical and programming experience equivalent to CIS 1600 and CIS 1210.

Fall

Mutually Exclusive: CIS 4500

1 Course Unit

**CIS 5510 Computer and Network Security**

This is an introduction to topics in the security of computer systems and communication on networks of computers. The course covers four major areas: fundamentals of cryptography, security for communication protocols, security for operating systems and mobile programs, and security for electronic commerce. Sample specific topics include: passwords and offline attacks, DES, RSA, DSA, SHA, SSL, CBC, IPSec, SET, DDoS attacks, biometric authentication, PKI, smart cards, S/MIME, privacy on the Web, viruses, security models, wireless security, and sandboxing. Students will be expected to display knowledge of both theory and practice through written examinations and programming assignments. The course assumes experience equivalent to CIS 1600 and CIS 2400.

Fall or Spring

Mutually Exclusive: CIS 4510

1 Course Unit

**CIS 5520 Advanced Programming**

The goals of this course are twofold: (1) to take good programmers and turn them into excellent ones, and (2) to introduce them to a range of modern software engineering practices, in particular those embodied in advanced functional programming languages. Four courses involving significant programming and a discrete mathematics or modern algebra course. Enrollment by permission of the instructor only.

Fall or Spring

1 Course Unit

**CIS 5530 Networked Systems**

This course provides an introduction to fundamental concepts in the design and implementation of networked systems, their protocols, and applications. Topics to be covered include: Internet architecture, network applications, addressing, routing, transport protocols, network security, and peer-to-peer networks. The course will involve written assignments, examinations, and programming assignments. Students will work in teams to design and implement networked systems in layers, from routing protocols, transport protocols, to peer-to-peer networks. Preparation for this course would include taking CIS 1210 or equivalent knowledge.

Fall or Spring

1 Course Unit

**CIS 5540 Programming Paradigms**

Achieving mastery in a new programming language requires more than just learning a new syntax; rather, different languages support different ways to think about solving problems. Not all programming languages are inherently procedural or object-oriented. The intent of this course is to provide a basic understanding of a wide variety of programming paradigms, such as logic programming, functional programming, concurrent programming, rule-based programming, and others. Prerequisites: CIS 1210 or equivalent.

Fall or Spring

1 Course Unit

**CIS 5550 Internet and Web Systems**

This course focuses on the challenges encountered in building Internet and web systems: scalability, interoperability (of data and code), security and fault tolerance, consistency models, and location of resources, services, and data. We will examine how XML standards enable information exchange; how web services support cross-platform interoperability (and what their limitations are); how to build high-performance application servers; how "cloud computing" services work; how to perform Akamai-like content distribution; and how to provide transaction support in distributed environments. We will study techniques for locating machines, resources, and data (including directory systems, information retrieval indexing, ranking, and web search); and we will investigate how different architectures support scalability (and the issues they face). We will also examine ideas that have been proposed for tomorrow's Web, and we will see some of the challenges, research directions, and potential pitfalls. An important goal of the course is not simply to discuss issues and solutions, but to provide hands-on experience with a substantial implementation project. This semester's project will be a peer-to-peer implementation of a Google-style search engine, including distributed, scalable crawling; indexing with ranking; and even PageRank. As a side-effect of the material of this course you will learn about some aspects of large-scale software development assimilating large APIs, thinking about modularity, reading other people's code, managing versions, debugging, etc. Prerequisite: Familiarity with threads and concurrency, strong Java programming skills.

Fall or Spring

Mutually Exclusive: CIS 4550

1 Course Unit

**CIS 5560 Cryptography**

This course is an introduction to cryptography, both theory and applications, intended for advanced undergraduates and graduate students. Topics covered include symmetric cryptography, message authentication, public-key cryptography, digital signatures, cryptanalysis, cryptographic security, and secure channels, as well as a selection of more advanced topics such as zero-knowledge proofs, secure multiparty computation, privacy-enhancing technologies, or lattice-based cryptography.

Fall or Spring

1 Course Unit

**CIS 5570 Programming for the Web**

This course will focus on web programming. The first half will focus on the basics of the internet and the Web, HTML and CSS, and basic and advanced Ruby. The second half will focus on Rails. Teams (of size 2-3) will build a web application in the second half of the semester as the class project. Through Rails, we'll explore the "culture" of web programming such as agile methodology, testing, key aspects of software engineering, using web services and APIs, and deploying to the cloud.

1 Course Unit

**CIS 5590 Programming and Problem Solving**

This course develops students problem solving skills using techniques that they have learned during their CS training. Over the course of the semester, students work on group projects in which they use programming techniques to solve open-ended problems, e.g. optimization, simulation, etc. There are no "correct" answers to these problems; rather, the focus is on the four steps of the problem solving process: algorithmic thinking; programming; analysis; and communication. Prerequisite: Proficiency in Java.

Fall or Spring

1 Course Unit

**CIS 5600 Interactive Computer Graphics**

This course focuses on programming the essential mathematical and geometric concepts underlying modern computer graphics. Using 3D interactive implementations, it covers fundamental topics such as mesh data structures, transformation sequences, rendering algorithms, and curve interpolation for animation. Students are also introduced to two programming languages widely used in the computer graphics industry: C++ and GLSL. The curriculum is heavily project-based, and culminates in a group project focused on building an interactive first-person world exploration application using the various real-time interaction and rendering algorithms learned throughout the semester.

Fall

1 Course Unit

**CIS 5610 Advanced Computer Graphics**

This course is designed to provide a comprehensive overview to computer graphics techniques in 3D modeling, image synthesis, and rendering. Topics cover: geometric transformations, geometric algorithms, software systems, 3D object models (surface, volume and implicit), visible surface algorithms, image synthesis, shading, mapping, ray tracing, radiosity, global illumination, sampling, anti-aliasing, Monte Carlo path tracing, and photon mapping. Prerequisite: A working knowledge of C++ programming is required (one year programming experience in general). Knowledge of vector geometry is useful.

Fall or Spring

Mutually Exclusive: CIS 4610

1 Course Unit

**CIS 5620 Computer Animation**

This course covers core subject matter common to the fields of robotics, character animation and embodied intelligent agents. The intent of the course is to provide the student with a solid technical foundation for developing, animating and controlling articulated systems used in interactive computer games, virtual reality simulations and high-end animation applications. The course balances theory with practice by "looking under the hood" of current animation systems and authoring tools and exams the technologies and techniques used from both a computer science and engineering perspective. Topics covered include: geometric coordinate systems and transformations; quaternions; parametric curves and surfaces; forward and inverse kinematics; dynamic systems and control; computer simulation; keyframe, motion capture and procedural animation; behavior-based animation and control; facial animation; smart characters and intelligent agents. Prerequisite: Previous exposure to major concepts in linear algebra (i.e. vector matrix math), curves and surfaces, dynamical systems (e.g. 2nd order mass-spring-damper systems) and 3D computer graphics has also been assumed in the preparation of the course materials.

Fall or Spring

Mutually Exclusive: CIS 4620

1 Course Unit

**CIS 5630 Physically Based Animation**

This course introduces students to common physically based simulation techniques for animation of fluids and gases, rigid and deformable solids, cloth, explosions, fire, smoke, virtual characters, and other systems. Physically based simulation techniques allow for creation of extremely realistic special effects for movies, video games and surgical simulation systems. We will learn state-of-the-art techniques that are commonly used in current special effects and animation studios and in video games community. To gain hands-on experience, students will implement basic simulators for several systems. The topics will include: Particle Systems, Mass spring systems, Deformable Solids & Fracture, Cloth, Explosions & Fire, Smoke, Fluids, Deformable active characters, Simulation and control of rigid bodies, Rigid body dynamics, Collision detection and handling, Simulation of articulated characters, Simulated characters in games. The course is appropriate for both upper level undergraduate and graduate students. Prerequisite: Students should have a good knowledge of object-oriented programming (C++) and basic familiarity with linear algebra and physics. Background in computer graphics is required (CIS 461 and 561).

Fall or Spring

1 Course Unit

**CIS 5640 Game Design and Development**

The intent of the course is to provide students with a solid theoretical understanding of the core creative principles, concepts, and game play structures/schemas underlying most game designs. The course also will examine game development from an engineering point of view, including: game play mechanics, game engine software and hardware architectures, user interfaces, design documents, playtesting and production methods. Basic understanding of 3D graphics and animation principles, prior exposure to scripting and programming languages such as Python, C and C++

Fall or Spring

1 Course Unit

**CIS 5650 GPU Programming and Architecture**

This course examines the architecture and capabilities of modern GPUs. The graphics processing unit (GPU) is orders of magnitude faster for computation than traditional CPU, and with the power of general purpose programming, GPUs can be used for a diverse set of applications far removed from their traditional graphics usage. In this course, students will learn to program and optimize GPUs for computationally intensive algorithms. Topics covered include architectural aspects of modern GPUs, with a special focus on massively parallel programming, writing programs using CUDA, and using the GPU for graphics and general purpose applications in the area of geometry modeling, physical simulation, scientific computing and games. Students are expected to have a basic understanding of computer architecture and graphics, and should be proficient in C/C++. This course is appropriate as an upper-level undergraduate CIS elective. Computer Graphics background is strongly recommended, which can be met with CIS 4600, CIS 5600, or an equivalent Computer Graphics course.

Fall

1 Course Unit

**CIS 5660 Procedural Computer Graphics**

Sprawling cities, dense vegetation, infinite worlds - procedural graphics empower technical artists to quickly create complex digital assets that would otherwise be unfeasible. This course is intended to introduce the mathematical and algorithmic foundations of procedural modeling and animation techniques, and to offer hands-on experience designing and implementing visual recipes in original graphics projects by applying these methods. Students should have a strong interest in both the creative and technical aspects of computer graphics, as well as a solid programming background.

Fall or Spring

Prerequisite: CIS 4600 OR CIS 5600

1 Course Unit

**CIS 5670 Scientific Computing**

This course will focus on numerical algorithms and scientific computing techniques that are practical and efficient for a number of canonical science and engineering applications. Built on top of classical theories in multi-variable calculus and linear algebra (as a prerequisite), the lectures in this course will strongly focus on explaining numerical methods for applying these mathematical theories to practical engineering problems. Students will be expected to implement solutions and software tools using MATLAB/C++, practice state-of-the-art parallel computing paradigms, and learn scientific visualization techniques using modern software packages. Prerequisites: MATH 2400; knowledge of C++, Python or MATLAB

Fall or Spring

Mutually Exclusive: CIS 4670

1 Course Unit

**CIS 5680 Game Design Practicum**

The objective of the game design practicum is to provide students with hands on experience designing and developing 3D games as well as Virtual Reality (VR) and Augmented Reality (AR) applications. Working in teams of three, students will gain experience brainstorming original game concepts, writing formal design documents and developing a fully functional prototype of their game/application. Technical features to be designed and implemented include novel game mechanics and/or user interaction models, game physics, character animation, game AI (i.e. movement control, path planning, decision making, etc.), sound effects and background music, 2D graphical user interface (GUI) design and multiplayer networking capabilities. State-of-the-art game engine middleware such as the Unity3D and Unreal game engines will be used to expose student to commercial-grade software, production methodologies and art asset pipelines. As a result of their game development efforts, students will learn first-hand about the creative process, design documentation, object-oriented software design and engineering, project management (including effective team collaboration and communication techniques), design iteration through user feedback and play-testing, and most importantly, what makes a game or virtual reality experience interesting and fun to play.

Fall, Spring, and Summer Terms

Prerequisite: CIS 4600 OR CIS 5600

1 Course Unit

**CIS 5690 GPU Computing for Machine Learning Systems**

In this course, we will explore massively parallel programming, specifically on graphics processing units (GPUs), with immediate application to machine learning (ML) and artificial intelligence (AI). We'll first outline computational aspects of ML and connect parallel programming to common components of deep learning. You will gain proficiency in GPU programming basics through hands-on projects with industry best practices and tools, eventually building up to implementing components of modern neural models. Recommended Prerequisites: ESE 5460. Course projects require knowledge of C/C++.

Prerequisite: CIT 5930 AND CIT 5940 AND CIT 5950 AND CIT 5960

1 Course Unit

**CIS 5710 Computer Organization and Design**

This is the second computer organization course and focuses on computer hardware design. Topics covered are: (1) basic digital system design including finite state machines, (2) instruction set design and simple RISC assembly programming, (3) quantitative evaluation of computer performance, (4) circuits for integer and floating-point arithmetic, (5) datapath and control, (6) micro-programming, (7) pipelining, (8) storage hierarchy and virtual memory, (9) input/output, (10) different forms of parallelism including instruction level parallelism, data-level parallelism using both vectors and message-passing multi-processors, and thread-level parallelism using shared memory multiprocessors.

Basic cache coherence and synchronization. Prerequisite: CIS 2400 or equivalent.

Spring

Mutually Exclusive: CIS 4710

1 Course Unit

**CIS 5730 Software Engineering**

Writing a "program" is easy. Developing a "software product", however, introduces numerous challenges that make it a much more difficult task.

This course will look at how professional software engineers address those challenges, by investigating best practices from industry and emerging trends in software engineering research. Topics will focus on software maintenance issues, including: test case generation and test suite adequacy; code analysis; verification and model checking; debugging and fault localization; refactoring and regression testing; and software design and quality. Prerequisite: CIS 3500 or equivalent.

Fall

1 Course Unit

**CIS 5800 Machine Perception**

An introduction to the problems of computer vision and other forms of machine perception that can be solved using geometrical approaches rather than statistical methods. Emphasis will be placed on both analytical and computational techniques. This course is designed to provide students with an exposure to the fundamental mathematical and algorithmic techniques that are used to tackle challenging image based modeling problems. The subject matter of this course finds application in the fields of Computer Vision, Computer Graphics and Robotics. Some of the topics to be covered include: Projective Geometry, Camera Calibration, Image Formation, Projective, Affine and Euclidean Transformations, Computational Stereopsis, and the recovery of 3D structure from multiple 2D images. This course will also explore various approaches to object recognition that make use of geometric techniques, these would include alignment based methods and techniques that exploit geometric invariants. In the assignments for this course, students will be able to apply the techniques to actual computer vision problems. This course is appropriate as an upper-level undergraduate CIS elective. A solid grasp of the fundamentals of linear algebra. Some knowledge of programming in C and/or Matlab

Fall

1 Course Unit

**CIS 5810 Computer Vision & Computational Photography**

This is an introductory course to Computer Vision and Computational Photography. This course will explore three topics: 1) image morphing, 2) image matching and stitching, and 3) image recognition. This course is intended to provide a hands-on experience with interesting things to do on images/videos. The world is becoming image-centric. Cameras are now found everywhere, in our cell phones, automobiles, even in medical surgery tools. Computer vision technology has led to latest innovations in areas such as Hollywood movie production, medical diagnosis, biometrics, and digital library. This course is suited for students from all Engineering backgrounds, who have the basic knowledge of linear algebra and programming, and a lot of imagination.

Fall

Mutually Exclusive: CIS 4810

1 Course Unit

**CIS 5970 Master's Thesis Research**

For students working on an advanced research leading to the completion of a Master's thesis.

Fall or Spring

1-2 Course Units

**CIS 5990 Independent Study for Masters Students**

For master's students studying a specific advanced subject area in computer and information science. Involves coursework and class presentations. A CIS 5990 course unit will invariably include formally gradable work comparable to that in a CIS 500-level course. Students should discuss with the faculty supervisor the scope of the Independent Study, expectations, work involved, etc.

Fall or Spring  
1-4 Course Units

**CIS 6010 Advanced Topics in Computer Architecture**

This course will focus on research topics in computer architecture, and include reading and presenting research papers and an optional project. The content will differ with each offering, covering topics such as multicore programmability, datacenter and warehouse-scale computing, security, energy-efficient architectures, etc.

Fall  
Prerequisite: CIS 5710 OR CIS 4710  
1 Course Unit

**CIS 6100 Advanced Geometric Methods in Computer Science**

The purpose of this course is to present some of the advanced geometric methods used in geometric modeling, computer graphics, computer vision, etc. The topics may vary from year to year, and will be selected among the following subjects (nonexhaustive list): Introduction to projective geometry with applications to rational curves and surfaces, control points for rational curves, rectangular and triangular rational patches, drawing closed rational curves and surfaces; Differential geometry of curves (curvature, torsion, osculating planes, the Frenet frame, osculating circles, osculating spheres); Differential geometry of surfaces (first fundamental form, normal curvature, second fundamental form, geodesic curvature, Christoffel symbols, principal curvatures, Gaussian curvature, mean curvature, the Gauss map and its derivative  $dN$ , the Dupin indicatrix, the Theorema Egregium equations of Codazzi-Mainardi, Bonnet's theorem, lines of curvatures, geodesic torsion, asymptotic lines, geodesic lines, local Gauss-Bonnet theorem).

Spring  
1 Course Unit

**CIS 6200 Advanced Topics in Machine Learning**

This course covers a variety of advanced topics in machine learning, such as the following: statistical learning theory (statistical consistency properties of surrogate loss minimizing algorithms); approximate inference in probabilistic graphical models (variational inference methods and sampling-based inference methods); structured prediction (algorithms and theory for supervised learning problems involving complex/structured labels); and online learning in complex/structured domains. The precise topics covered may vary from year to year based on student interest and developments in the field.

Fall  
Prerequisite: CIS 5200  
1 Course Unit

**CIS 6250 Theory of Machine Learning**

This course is an introduction to the theory of Machine Learning, a field which attempts to provide algorithmic, complexity-theoretic and statistical foundations to modern machine learning. The focus is on topics in machine learning theory for researchers and students in artificial intelligence, neural networks, theoretical computer science, and statistics.

Fall or Spring  
1 Course Unit

**CIS 6300 Advanced Topics in Natural Language Processing**

Different topics selected each offering; e.g., NL generation, question-answering, information extraction, machine translation, restricted grammar formalisms, computational lexical semantics, etc.

Fall or Spring  
Prerequisite: CIS 5300  
1 Course Unit

**CIS 6400 Advance Topics in Software Systems: Data Driven IoT/Edge Computing**

This course is to explore selected topics in data driven IoT/Edge Computing. We are currently witnessing a technological paradigm shift, in which the IoT systems are increasingly deployed in society. This course is to study emerging paradigms in IoT/Edge Computing and to learn how to develop data driven applications that can harness the power of the IoT/Edge computing. For application domains, the course will target connected medical devices, smart home for aging, and connected automotive systems. Topics to be covered include IoT/Edge computing architectures, the Internet of medical things, connected vehicles, anomaly detection, mixed initiative systems, closed-loop systems, resource allocation, and security and privacy. The course will require a significant term project in connected health or connected automotive domains.

Spring  
Prerequisite: (CIS 5450 OR CIS 5190) AND (CIS 5050 or CIS 5410)  
1 Course Unit

**CIS 6500 Advanced Topics in Databases**

Advanced topics in databases: distributed databases, integrity constraints, failure, concurrency control, relevant relational theory, semantics of data models, the interface between programming of languages and databases. Object-oriented databases. New topics are discussed each year.

Fall  
Prerequisite: CIS 5500 OR CIS 4500  
1 Course Unit

**CIS 6600 Advanced Topics in Computer Graphics and Animation**

The goal of the course is to review state-of-the art research in the fields of computer graphics and animation as well as provide students with working knowledge of how to convert theory to practice by developing an associated graphics/animation authoring tool. The course is comprised of primers, lectures, student presentations and the authoring tool group project. Each student will be responsible for presenting one primer and at least two SIGGRAPH papers to the class. Working in teams of two, students will design and develop an authoring tool that facilitates the creation of a new type of user interaction, animation/simulation capability or 3D graphics special effect. Research papers published in the SigGraph Conference proceedings will provide the basis for the features/functionality/special effects that can be selected for implementation in the authoring tool. Each group will analyze the need and user requirements for the tool they plan to develop, prepare a formal software design document, construct a project work plan, develop the authoring tool functionality and user interface, test the design and demonstrate the authoring of associated content. A plug-in to standard authoring tools such as Maya or Houdini must also be developed to enable importing of appropriate assets and/or exporting of results.

Spring  
Prerequisite: CIS 5600  
1 Course Unit



**CIS 6700 Advanced Topics in Programming Languages**

The details of this course change from year to year, but its purpose is to cover theoretical topics related to programming languages. Some central topics include: denotational vs operational semantics, domain theory and category theory, the lambda calculus, type theory (including recursive types, generics, type inference and modules), logics of programs and associated completeness and decidability problems, specification languages, and models of concurrency. The course requires a degree of mathematical sophistication.

Fall or Spring

Prerequisite: CIS 5000

1 Course Unit

**CIS 6730 Computer-Aided Verification**

This course introduces the theory and practice of formal methods for the design and analysis of concurrent and embedded systems. The emphasis is on the underlying logical and automata-theoretic concepts, the algorithmic solutions, and heuristics to cope with the high computational complexity. Topics: Models and semantics of reactive systems; Verification algorithms; Verification techniques. Topics may vary depending on instructor. Basic knowledge of algorithms, data structures, automata theory, propositional logic, operating systems, communication protocols, and hardware (CIS 2620, CIS 3800, or permission of the instructor).

Fall or Spring

1 Course Unit

**CIS 6770 Advanced Topics in Algorithms and Complexity**

This course covers various aspects of discrete algorithms. Graph-theoretic algorithms in computational biology, and randomization and computation; literature in dynamic graph algorithms, approximation algorithms, and other areas according to student interests. Consent of the instructor.

Fall

1 Course Unit

**CIS 6800 Advanced Topics in Machine Perception**

Graduate seminar in advanced work on machine perception as it applies to robots as well as to the modeling of human perception. Topics vary with each offering. A previous course in machine perception or knowledge of image processing, experience with an operating system and language such as Unix and C, and aptitude for mathematics.

Spring

1 Course Unit

**CIS 6820 Friendly Logics**

The use of logical formalisms in Computer Science is dominated by a fundamental conflict: expressiveness vs. algorithmic tractability. Database constraint logics, temporal logics and description logics are successful compromises in this conflict: (1) they are expressive enough for practical specifications in certain areas, and (2) there exist interesting algorithms for the automated use of these specifications. Interesting connections can be made between these logics because temporal and description logics are modal logics, which in turn can be seen, as can database constraint logics, as certain fragments of first-order logic. These connections might benefit research in databases, computer-aided verification and AI. Discussion includes other interesting connections, eg., with SLD-resolution, with constraint satisfaction problems, with finite model theory and with automata theory.

Fall or Spring

1 Course Unit

**CIS 7000 Special Topics**

One time course offerings of special interest. Equivalent to a CIS 5XX level course.

Not Offered Every Year

1 Course Unit

**CIS 7980 Explaining Explanation**

In the social sciences we often use the word "explanation" as if (a) we know what we mean by it, and (b) we mean the same thing that other people do. In this course we will critically examine these assumptions and their consequences for scientific progress. In part 1 of the course we will examine how, in practice, researchers invoke at least three logically and conceptually distinct meanings of "explanation": identification of causal mechanisms; ability to predict (account for variance in) some outcome; and ability to make subjective sense of something. In part 2 we will examine how and when these different meanings are invoked across a variety of domains, focusing on social science, history, business, and machine learning, and will explore how conflation of these distinct concepts may have created confusion about the goals of science and how we evaluate its progress. Finally, in part 3 we will discuss some related topics such as null hypothesis testing and the replication crisis. We will also discuss specific practices that could help researchers clarify exactly what they mean when they claim to have "explained" something, and how adoption of such practices may help social science be more useful and relevant to society.

Also Offered As: COMM 8980, OIDD 9530

1 Course Unit

**CIS 8000 PhD Special Topics**

One-time course offerings of special interest. Equivalent to CIS seminar course. Offerings to be determined.

Fall or Spring

1 Course Unit

**CIS 8100 Writing and Speaking with Style**

This course is aimed at training CIS PhD students to excel in writing and presenting research results.

Fall or Spring

1 Course Unit

**CIS 8950 Teaching Practicum**

Enrollment for students participating in Teaching Practicum.

Fall or Spring

1 Course Unit

**CIS 8990 Doctoral Independent Study**

For doctoral students studying a specific advanced subject area in computer and information science. The Independent Study may involve coursework, presentations, and formally gradable work comparable to that in a CIS 5000 or 6000 level course. The Independent Study may also be used by doctoral students to explore research options with faculty, prior to determining a thesis topic. Students should discuss with the faculty supervisor the scope of the Independent Study, expectations, work involved, etc. The Independent Study should not be used for ongoing research towards a thesis, for which the CIS 9990 designation should be used.

Fall or Spring

1-4 Course Units

**CIS 9950 Dissertation**

For Ph.D. candidates working exclusively on their dissertation research, having completed enrollment for a total of ten semesters (fall and spring). There is no credit or grade for CIS 9950.

0 Course Units

**CIS 9990 Thesis/Dissertation Research**

For students pursuing advanced research to fulfill PhD dissertation requirements.

Fall or Spring

1-4 Course Units