SCS Concentrations

The School of Computer Science (SCS) offers concentrations for SCS students in various aspects of computing to provide greater depth to their education. Computer Science majors can substitute an SCS concentration for the minor requirement. Artificial Intelligence and Computational Biology majors can complete an SCS concentration if they wish, but it is not required for these degrees.

Note: At the present time, concentrations are not shown on official transcripts.

SCS Concentrations are currently available to SCS students only and assume these students have taken most/all of the CS core: 15-122/15-150/15-210/15-213/15-251. For SCS students entering in Fall 2018 or later, these students may not pursue a minor within SCS; instead they can pursue the related concentration. For example, instead of pursuing the Software Engineering minor, these SCS students could pursue the new Software Engineering concentration.

Consult the SCS undergraduate concentrations website (https://www.cs.cmu.edu/undergraduate-concentrations/) for information about these concentrations as they are approved. For SCS students, consult with your academic advisor for more information about available concentrations and requirements.

- Algorithms and Complexity
- Computational Biology
- Computer Graphics
- Computer Systems
- Human-Computer Interaction
- Language Technologies
- Machine Learning
- Principles of Programming Languages
- Robotics
- Security and Privacy
- Software Engineering

Algorithms and Complexity Concentration

This concentration is available to SCS students only.

Ryan O'Donnell, *Concentration Director*
Location: GHC 7213

Amy Weis, *Concentration Coordinator*
Location: GHC 4115

The goal of the Algorithms and Complexity concentration is to give SCS students a deep background in the theory of computation as it relates to algorithms and computational complexity. The expectation is that students who complete this concentration will have the background to pursue topics at the PhD level at any top program in the country. Furthermore we expect the reasoning skills gained as part of this concentration could be a significant help in a wide variety of positions in industry.

The concentration is designed to be reasonably flexible covering a wide area of topics within the area of algorithms and complexity. This includes central topics within the area such as complexity theory, and algorithms, but also includes theory as used in areas such as Computational Geometry, Graph Theory, Cryptography, Machine Learning, Algorithms for Large Data, Error Correcting Codes, and Parallel Algorithms.

Common themes of all courses covered by the concentration are the following:

- Clearly defined formalisms of the subject matter.
- A substantial component involving rigorous mathematical analysis, including proofs.
- Abstracting away from specific applications to a more general context.
- Relating algorithms and/or complexity of computation to a variety of complexity measures such as time, space, communication, or information content.

Any given course does not have to exclusively cover these themes and can, for example, also cover experimental aspects of algorithms, or examples applied to quite specific applications.

Learning Objectives

We do not expect students to have high proficiency in all the examples listed, but to gain at least some proficiency from each category.

- The ability to take a loosely defined problem and clearly pose it as a well defined problem specification.
- The understanding of several advanced algorithms beyond what is covered in the core.
- The appreciation a variety of models for bounding resources, such as information theory, space complexity, parallel complexity, communication complexity, proof complexity, query complexity, and hardness of approximation.
- The ability to understand and apply a variety of advanced algorithmic techniques and proof techniques, such as Lovasz Local Lemma, Johnson Lindenstrauss, Chernoff Bounds, sparsification, expanders, probabilistic method, regret bounds, spectral graph theory, fixed parameter tractability and semi-indefinite programming.
- The ability to recognize flaws in ill-formed proofs.
- The ability to formulate new questions about the field.

Prerequisites

The following courses must be completed before the concentration can be completed:

15-210	Parallel and Sequential Data Structures and Algorithms	12
15-251	Great Ideas in Theoretical Computer Science	12
15-259	Probability and Computing	12
or 21-325	Probability	
or 36-218	Probability Theory for Computer Scientists	
15-451	Algorithm Design and Analysis	12

It is expected that all students will start the concentration after having finished all but 15-451.

Course Requirements

The curriculum consists of one required course and at least three elective courses. The three elective courses must sum to at least 30 units. The elective courses will vary from year to year.

Required:

15-455	Undergraduate Complexity Theory	9

Electives (at least three courses with a total of 30 units or more):

15-354	Computational Discrete Mathematics	12
15-356	Introduction to Cryptography	12
15-458	Discrete Differential Geometry	12
15-459	Undergraduate Quantum Computation	9
15-751	A Theorist's Toolkit	12
21-301	Combinatorics	9
21-484	Graph Theory	9
Special permission required:		
47-834	Linear Programming	6
47-835	Network Optimization I	6
47-836	Network Optimization II	6
Other graduate-level courses as approved by the concentration director		

Students can apply one semester of a senior honors research thesis or research-based independent study in a topic related to this concentration, as approved by the concentration director/advisor, as one of the elective courses for this concentration. This research must have a significant communication component, including a paper or technical report, and a poster presentation. Any research course can count for at most 12 units toward the concentration and can count for at most one elective.

The choice of available elective courses will be posted prior to registration each semester.

Double Counting

The concentration will require that 3 courses (at least 27 units) are not double counted with any other requirements of any Major, Minor, or other concentration the student is pursuing.

Advising and Management

Courses in the list of electives will be approved by the director on a yearly basis under consultation of the algorithms and complexity group (to help evaluate the relevance of the courses) and the CS Program Director (to help flag any logistical issues). Any special requests by a student for counting a course out of the list, will go to the director. The director will also approve any research units.

Students interested in this concentration should contact the concentration director for an initial advising consultation.

Computational Biology Concentration

This concentration is available to SCS students only.

Phillip Compeau, *Concentration Director*
Location: GHC 7403

Tara Seman, *Concentration Coordinator*
Location: GHC 7414

The general goal of the Computational Biology Concentration is to provide foundational coursework in computational biology that will allow undergraduate students in the Carnegie Mellon University School of Computer Science to start building a skillset useful for understanding many of the modern technologies developed by researchers as well as companies in the biotech and biomedical arenas.

This concentration consists of four core courses providing breadth in computational biology across laboratory methods, machine learning, genomics, and modeling of biological systems, as well as one elective that allows students to complete depth coursework in an area of interest, including undergraduate research.

Learning Objectives

Students will, by way of completing this concentration:

- model biological systems at the molecular and cellular levels using a variety of approaches;
- generate their own high throughput molecular biology data in a laboratory setting, and apply computational techniques to analyze the data they generate;
- transform hazy biological problems involving genomic data into well-defined computational problems, design algorithms to solve these problems, and adapt them to biological data;
- explore additional coursework of interest in genomics, biological research automation, biological image analysis, or computational biology research.

This concentration also provides students completing a computational degree other than the major in computational biology (http://www.cbd.cmu.edu/education/bs-in-computational-biology/) with the opportunity to make a transition toward a career in computational biology. We have compiled information on over 250 companies working on computational biology into a unique web resource for students both inside and outside of Carnegie Mellon (http://careers.cbd.cmu.edu (http://careers.cbd.cmu.edu/)). These companies work on diverse topics from the automation of biological research to drug discovery to wearable medical devices to genetic diagnostics. Increasingly, when we interact with these companies, they want computationally minded candidates with as much knowledge of standard approaches in computational biology as possible.

Prerequisites

Note that not all of the prerequisites below are required to take every course in this concentration (for example, 02-251 does not have any of the pre-requisites below), but these courses are required to complete all of the required coursework and should be completed early within this concentration.

15-122	Principles of Imperative Computation	12
15-151	Mathematical Foundations for Computer Science	12
15-210	Parallel and Sequential Data Structures and Algorithms	12
21-241	Matrices and Linear Transformations	11
36-218	Probability Theory for Computer Scientists	9

Further, the following two courses are not technically required as prerequisites to the courses in this concentration, but they are strongly suggested prerequisites because they provide students with helpful surveys of fundamental topics in biology and computational biology.

02-251	Great Ideas in Computational Biology	12
03-151	Honors Modern Biology	10
or 03-121	Modern Biology	

Requirements

Five courses in total are required for this concentration. The following four courses are required as part of a central core of coursework; they consist of three computational biology courses as well as an introductory machine learning course, which today is fundamental for even an introductory understanding of the field.

		Units
02-261	Quantitative Cell and Molecular Biology Laboratory (03-343, Experimental Techniques in Molecular Biology, may be taken if 02-261 is not offered)	Var.
02-510	Computational Genomics	12
02-512	Computational Methods for Biological Modeling and Simulation	9
10-315	Introduction to Machine Learning (SCS Majors)	12

In addition to these four courses above, one elective course is required. Any 02-listed (Computational Biology Department) undergraduate course of at least 9 units at the 300-level or above may satisfy this requirement; graduate courses may be applied to this category with permission. The Computational Biology Department is growing quickly, but at the time of writing, the courses that are regularly offered by the department that would satisfy this requirement are the following:

02-317	Algorithms in Nature	9
02-319	Genomics and Epigenetics of the Brain	9
02-425	Computational Methods for Proteogenomics and Metabolomics	9
02-450	Automation of Scientific Research	9
02-499	Independent Study in Computational Biology	Var.
02-500	Undergraduate Research in Computational Biology (03-441/03-541 may be taken if 02-500 is not offered)	Var.
02-514	String Algorithms	12
02-515	Advanced Topics in Computational Genomics	12
02-518	Computational Medicine	12

Students can apply one semester of a senior honors research thesis or research-based independent study in a topic related to this concentration, as approved by the concentration director/advisor, as the elective course for this concentration. This research must have a significant communication component, including a paper or technical report, and a poster presentation. Any research course can count for at most 12 units toward the concentration and can count for at most one elective.

Double Counting

At most two courses can double count with all program requirements for majors, minors and other concentrations being pursued by the student. Courses used as free electives for a major are not considered double counted.

Accordingly, this concentration is expressly closed to majors and additional majors in computational biology.

CS and AI majors completing this concentration are encouraged to double-count 10-315 as well as 02-261 as their lab science course. Suggested prerequisites 03-151 and 02-251 also count as requirements for these degrees (as a Science & Engineering course and CS Domains course, respectively).

Advising and Management

The day-to-day management of this concentration (including declaration of the concentration, exception requests, overseeing student audits, advising, etc.) is handled by Phillip Compeau, Assistant Department Head in the Computational Biology Department. Administrative support for the concentration is provided by Samantha Mudrinich. Curricular organization and annual review will be managed by the Computational Biology Undergraduate Review Committee.

SCS students interested in this concentration should set up an appointment with Phillip Compeau for a brief interview.

Computer Graphics Concentration

This concentration is available to SCS students only.

Nancy Pollard, *Concentration Director*
Location: EDSH 227

Amy Weis, *Concentration Coordinator*
Location: GHC 4115

The SCS Computer Graphics Concentration provides an opportunity for SCS undergraduate students at Carnegie Mellon to learn Computer Graphics foundations and specialties from a variety of application and research areas. Students gain a broad view of Computer Graphics in an introductory course and in-depth experience from a choice of topic areas, including the option of independent research. Courses include computational photography, computer games, computer animation, computational geometry, and physics-based rendering. This concentration provides an excellent introduction to the area for students considering industry and the opportunity for research experience for those considering graduate study.

Learning Objectives

Students completing this concentration will be able to demonstrate the following skills and learning:

- Create geometric representations for objects and natural phenomena.
- Develop software systems that work with these representations to capture, manipulate, and create artistic and real-world scenes.
- Discuss the pros and cons of alternative models of light transport.

- Develop software systems that utilize these models to create graphical images.
- State the key principles of animation and computer game development.
- Work with kinematic and physically simulated representations of motion to create animations of characters and natural phenomena.
- List key features of camera design, including for the modern camera.
- Work with images and video to perform common operations or generate new creative content.
- Execute common mathematical operations of computer graphics, including geometric transformations, projections, sampling algorithms, modeling of physical processes, optimization, and behavior-based modeling.
- Produce event-driven programs of moderate to large size with constantly evolving graphical / visual display.
- Sketch the layout of a typical massively parallel graphics processing unit (GPU).
- Write software to program the GPU for real-time graphics tasks.

Prerequisites

The following courses must be completed before the concentration can be completed:

15-213	Introduction to Computer Systems	12
21-241	Matrices and Linear Transformations	11
21-259	Calculus in Three Dimensions	10

It is expected that all students will start the concentration after having finished all but 15-451.

Course Requirements

The curriculum consists of one required course and at least three elective courses. The three elective courses must sum to at least 30 units. The elective courses will vary from year to year.

Required:

15-462	Computer Graphics	12

Electives (at least three courses with a total of 30 units or more):

15-365	Experimental Animation	12
15-418	Parallel Computer Architecture and Programming	12
15-456	Computational Geometry	9
15-458	Discrete Differential Geometry	12
15-463	Computational Photography	12
15-464	Technical Animation	12
15-465	Animation Art and Technology	12
15-466	Computer Game Programming	12
15-468	Physics-Based Rendering	12
15-469	Special Topic: Visual Computing Systems	12
Special permission required:		
16-726	Learning-based Image Synthesis	12
16-823	Physics-based Methods in Vision (Appearance Modeling)	12

Students can apply one semester of a senior honors research thesis or research-based independent study in a topic related to this concentration, as approved by the concentration director/advisor, as one of the elective courses for this concentration. This research must have a significant communication component, including a paper or technical report, and a poster presentation. Any research course can count for at most 12 units toward the concentration and can count for at most one elective.

RELATION TO OTHER PROGRAMS

The Computer Graphics Concentration overlaps slightly in course offerings of the IDeATE minors in Game Design and Animation and Special Effects, and interested students may also want to investigate these minors. What distinguishes the Computer Graphics Concentration is its focus on the fundamental theory, algorithms, and systems that are relevant to graphics. As such, this is an ideal choice for students who wish to get more in-depth, e.g., in preparation for graduate study or a career in Computer Graphics research.

Other courses to be aware of include:

- 21-344 Numerical Linear Algebra
- 10-725 Convex Optimization

Although they are not a core part of the concentration, these can be valuable courses for additional preparation for the mathematics that runs throughout computer graphics, as well as to prepare for future research and exploration in the field.

Double Counting

At most one course taken for the concentration may be double counted towards any major, minor or other concentration being pursued by the student. No other double counting is permitted.

Computer Systems Concentration

This concentration is available to SCS students only.

Brian Railing, *Concentration Director and Advisor*
Location: GHC 6005

Amy Weis, *Concentration Coordinator*
Location: GHC 4115

The goal of the Computer Systems concentration is to give students a broad background in the practical understanding of designing and building systems. Students who complete this concentration are expected to be able to both pursue topics at the Ph.D. level at top programs, as well as industry work, either applying these concepts or directly working within the areas of kernel development, compiler improvements, designing distributed systems, et cetera.

The concentration is designed to be flexible in covering the wide area of systems topics. Two courses from the Computer Science major's "systems" constrained elective are required (List A below). The other courses come from a larger list of related courses (List B below). A limited amount of research credit can count toward the requirements.

Typically, systems courses include three aspects:

- A systems course educates students about how a class of computer systems works, both at a conceptual level and in practice. This includes the study of concrete problems faced in building a particular class of systems and successful solutions to these problems.
- Systems courses address how properties of modern hardware (e.g., processors, net- works, storage hierarchies) influence the design and implementation of a class of soft- ware systems. This typically includes reasoning about concurrency, and understanding and measuring performance.
- To solidify the key systems organization principles, there is a significant project/system implementation aspect to the course, both to reinforce understanding of how these systems work, and to learn system building skills (i.e., not just programming, but also design, debugging, testing, etc.). The size of the programming tasks is course dependent, but a significant fraction of the course grade (e.g., at least 40%) is derived from project work.

Learning Objectives

Students completing this concentration will be able to demonstrate the following skills and learning:

- Students will be able to design, develop and deploy large computer systems and justify their design decisions.
- Students will synthesize the interaction and tradeoffs between different layers and components in computer systems.
- Students will demonstrate debugging expertise on complex and diverse bugs and issues during software development.
- Students will recognize diverse granularities of parallelism, apply them toward solving problems, and implement solutions that achieve correct execution while accounting for reliability, fault tolerance, performance, security, and scalability.

Prerequisites

All students will start the concentration after having finished 15-213 Introduction to Computer Systems (or its cross-listed equivalents) with a C or better, as 15-213 is a prerequisite either directly or indirectly for all courses in the concentration.

Course Requirements

The curriculum will consist of at least four courses: two courses from List A and at least two elective courses from List B. The courses taken from list A and B must sum to at least 51 units. The courses in List A will follow the Systems constrained elective list as part of the degree requirements for a B.S. in Computer Science. The elective courses on List B may vary from year to year, with a plan to review these requirements every three years.

List A (select two):

15-410	Operating System Design and Implementation	15
15-411	Compiler Design	15
15-418	Parallel Computer Architecture and Programming	12
15-440	Distributed Systems	12

15-441	Networking and the Internet	12
15-445	Database Systems	12

List B (select at least two):

Anything from List A, plus...		
15-319	Cloud Computing	12
15-330	Introduction to Computer Security	12
15-346	Computer Architecture: Design and Simulation	12
15-348	Embedded Systems	9
15-412	Operating System Practicum	Var.
15-415	Database Applications	12
17-422	Building User-Focused Sensing Systems	12
18-341	Logic Design and Verification	12
18-344	Computer Systems and the Hardware-Software Interface	12
18-349	Introduction to Embedded Systems	12
18-447	Introduction to Computer Architecture	12
Special permission required for:		
15-719	Advanced Cloud Computing	12
Other graduate-level courses in CSD as approved by the concentration director		

Students can apply one semester of a senior honors research thesis or research-based independent study in a topic related to this concentration, as approved by the concentration director/advisor, as one of the elective courses for this concentration in list B. This research must have a significant communication component, including a paper or technical report, and a poster presentation. Any research course can count for at most 12 units toward the concentration and can count for at most one elective.

Double Counting

The concentration will require that 3 courses (at least 27 units) are not double counted with any other requirements of any major, minor, or other concentration.

Advising and Management

The courses on the list of electives will be reviewed yearly by the concentration advisor through consultation with the Systems group (to help evaluate the relevance of the courses) and the CS Program Director (to help flag any logistical issues). Any special requests by a student for counting a course outside of the list will go to the concentration advisor. The concentration advisor will also approve any research units.

Students interested in this concentration should set up an initial advising consultation with the concentration advisor.

Human-Computer Interaction Concentration

This concentration is available to SCS students only.

Haiyi Zhu, *Undergraduate Director*
Email: haiyiz@andrew.cmu.edu

Amelia Baisley, *Concentration Manager*
Location: Newell-Simon Hall 3526B

In this concentration, students learn techniques, processes, principles, and theory of Human-Computer Interaction (HCI). This interdisciplinary field aims at understanding how interactions with digital technologies and services can augment what humans do. It also aims at understanding what design, prototyping, and evaluation processes lead to innovative digital technologies and services that fulfill human needs. The concentration enhances what is learned in the SCS majors by addressing how digital products and services can be designed and evaluated so they benefit individuals, small groups, organizations, larger networks, and societies. It is synergistic with SCS majors in that envisioning, designing, and implementing innovative digital interactions benefit from superior technical skill. The concentration consists of 5 courses (2 required courses and 3 electives).

The concentration helps prepare students for jobs as technically-skilled specialists in design and development of interactive systems. The concentration will give students a broader perspective on how technologies impact humans, which may help them move faster into product management positions. It also lays a foundation for graduate study in the field of Human-Computer Interaction.

Learning Objectives

Students will learn skills and methods for:

- Eliciting and understanding human objectives, preferences, and needs through qualitative and quantitative methods for data collection and analysis
- Generating and imagining possible solutions and design concepts that involve human/technology partnerships
- Basic visual design, including typography, grids, color and the use of images
- Design of interactive systems, experiences, and technologies
- Developing and evaluating interactive prototypes as a way of iteratively refining designs
- Evaluating interactive technologies to assess and improve their functioning through data-driven redesign, including discount and empirical evaluation methods

Students will also learn about:

- Digital technologies, including, possibly, web and mobile platforms, conversational technologies, wearable computing, gadgets, digital fabrication, virtual reality and mixed reality
- Human psychology, regarding individuals, groups, organizations, societies, and cultures, as it relates to interactions with digital products and services

Prerequisites

For this concentration, students should have completed the following courses prior to starting the concentration:

15-122	Principles of Imperative Computation	12
15-150	Principles of Functional Programming	12

Course Requirements

The following requirements are for students entering Fall 2023.

Students in an SCS major wanting to complete a concentration in HCI must complete 5 courses, namely, 2 required courses and 3 electives.

Required courses (2 courses)

05-391	Designing Human Centered Software	12
05-360	Interaction Design Fundamentals	12

Electives (3 courses)

1. At least one of the electives must have strong technical content and must be selected from the following list:

05-318	Human AI Interaction	12
05-333	Gadgets, Sensors and Activity Recognition in HCI	12
05-434	Machine Learning in Practice	12
05-499	Special Topics in HCI	12
05-839	Interactive Data Science	12
10-315	Introduction to Machine Learning (SCS Majors)	12
11-411	Natural Language Processing	12
15-237	Special Topic: Cross-Platform Mobile Web Apps	12
15-281	Artificial Intelligence: Representation and Problem Solving	12
15-365	Experimental Animation	12
15-388	Practical Data Science	9
15-462	Computer Graphics	12
15-464	Technical Animation	12
15-466	Computer Game Programming	12
15-494	Cognitive Robotics: The Future of Robot Toys	12
16-467	Human Robot Interaction	12
17-422	Building User-Focused Sensing Systems	12
17-428	Machine Learning and Sensing	12
17-437	Web Application Development	12
17-537	Artificial Intelligence Methods for Social Good	9
Other courses as approved by the concentration director		

2. At least one of the electives must have strong design content and must be selected from the following list:

05-361	Advanced Interaction Design	12
05-291	Learning Media Design	12
05-315	Persuasive Design	12
05-317	Design of Artificial Intelligence Products	12
05-418	Design Educational Games	12
05-452	Service Design	12
05-470	Digital Service Innovation	12

*Some (but not all) special topics classes (05-499) might also count towards the design elective requirement. Please consult with an HCI undergraduate advisor.

3. The remaining elective must be a course in HCI offered by the Human-Computer Interaction Institute, meaning it has a 05 number, or is included in the pre-approved list of electives maintained on the HCII website: https://hcii.cmu.edu/academics/hci-undergrad/electives (https://hcii.cmu.edu/academics/hci-undergrad/electives/)

Students can apply one semester of a senior thesis or research-based independent study (05-589) in a topic related to this concentration, as approved by the concentration director/advisor, as an elective courses for this concentration. This research must have a significant communication component, including a paper or technical report, and a poster presentation. Any research course can count for at most 12 units towards the concentration and can count for at most one elective.

Double Counting

At most 2 courses can be double counted with any major, minor or other concentration being pursued by the student.

ADVISING AND MANAGEMENT

Management will fall on the HCII Undergraduate Director and the HCII Academic Program Coordinator. The requirements for the courses will be reviewed annually by the HCII Curriculum Committee, in consultation with the URC.

Students in the HCI concentration will be advised by the HCII Undergraduate Director and/or the HCII Academic Program Coordinator, who also oversee and direct the HCI additional major and the HCI minor.

SCS Students interested in this concentration should contact the HCII Academic Program Coordinator for an initial advising consultation.

Language Technologies

This concentration is available to SCS Students only.

Carolyn P Rose (cp3a@andrew.cmu.edu), Concentration Director/Advisor
Location: 5415 Gates Hillman Center

Introduction

Human language technologies have become an increasingly central component of computer science. Information retrieval, machine translation and speech technology are used daily by the general public, while text mining, natural language processing and language-based tutoring are common within more specialized professional or educational environments. The Language Technologies Institute prepares students for this world by offering a concentration for SCS majors that gives these students the opportunity to not only learn about language technologies, but to also apply that knowledge through a directed project.

Learning Objectives

Students should be able to…

- Apply the principles of phonetics, morphology, morphosyntax, syntactic formalisms, discourse analysis and pragmatics and sociolinguistics to analyze, describe and discuss languages that they do not normally speak.
- Implement algorithms for organizing and searching large text collections through data mining or building text-based search engines.
- Process natural language input computationally to create automated, intelligent systems to perform language translation, implement question answering, build conversation agents, and query databases using natural interfaces.
- Develop systems that recognize human speech and synthesize human speech.
- Describe the limitations of speech and language analysis systems to pick the most appropriate tools and techniques for the given problem.
- Apply principles in language technologies to a research problem or study to advance the state of the art in this field.

Course Requirements

The Language Technologies Concentration requires that SCS students complete one core course and their choice of three elective courses of at least 9 units each. The electives can be chosen from a specific set of stand-alone courses. In addition to the four courses, students are required to do an undergraduate research project for at least 9 units to complete their concentration.

Core Course (12 units)

Students must take the core course below:

11-324	Human Language for Artificial Intelligence	12

Electives (minimum of 27 units)

Students must complete 3 electives from the following list of courses for a minimum of 27 units.

11-411	Natural Language Processing	12
11-441	Machine Learning for Text and Graph-based Mining	9
11-442	Search Engines	9
11-492	Speech Processing	12
11-344	Machine Learning in Practice	12
11-711	Advanced Natural Language Processing	12
11-731	Machine Translation and Sequence-to-Sequence Models	12
11-737	Multilingual Natural Language Processing.	12
11-747	Neural Networks for NLP	12
11-751	Speech Recognition and Understanding	12
11-761	Language and Statistics	12
11-776	Multimodal Affective Computing	12
80-180	Nature of Language	9

Research or Additional Elective (MINIMUM OF 9 UNITS)

Students can apply one semester of a senior honors research thesis or research-based independent study in a topic related to this concentration, as approved by the concentration director/advisor, as one of the elective courses for this concentration. This research must have a significant communication component, including a paper or technical report, and a poster presentation. Any research course can count for at most 12 units toward the concentration and can count for at most one elective. Students who opt not to do research may take an additional elective from the list above.

Prerequisites

The following courses are expected to be completed before the Core courses in the ML Concentration:

- CS background:
  - 15-122 Principles of Imperative Computation
  - 15-150 Principles of Functional Programming
- One year of calculus: 21-120 Differential and Integral Calculus and 21-122 Integration and Approximation or equivalent
- Matrix Algebra: 21-241 Matrices and Linear Transformations or 21-242 Matrix Theory
- Probability: 15-259 Probability and Computing or 21-325 Probability or 36-218 Probability Theory for Computer Scientists or 36-225 Introduction to Probability Theory

Double Counting

Courses being used to satisfy the requirements for the Language Technologies Concentration may not be counted towards another minor or concentration. 11-324 Human Language for Artificial Intelligence and 80-180 Nature of Language are the only courses that may be double counted with any other major or minor. All other courses used toward the concentration cannot be counted toward any requirement other than free electives.

Advising

The Language Technologies Concentration Advisor will assist students in selecting appropriate courses and reviewing completion of concentration requirements. Students working towards this concentration are welcome to reach out to the Director at any time for meetings to discuss the program, research directions, and career path questions.

Management

A student working towards the Language Technologies Concentration will select a Language Technologies faculty member as an Advisor with the assistance of the Director. There is also a Language Technologies Concentration Program Coordinator who is a staff member in the Language Technologies Institute. Curriculum changes to this concentration will be reviewed by the Undergraduate Review Committee for approval.

Machine Learning Concentration

This concentration is available to SCS students only.

Matt Gormley, *Concentration Director/Advisor*
Location: GHC 8103

Laura Winter, *Concentration Coordinator*
Location: GHC 8001

ml-concentration@cs.cmu.edu

Machine learning and statistical methods are increasingly used in many application areas including natural language processing, speech, vision, robotics, and computational biology. The Concentration in Machine Learning allows undergraduates to learn about the core principles of this field. The Concentration requires five courses (two core courses and three electives) from the School of Computer Science (SCS) and the Department of Statistics & Data Science. The electives primarily focus on core machine learning skills that could be broadly applicable to either industry or graduate work. A CS Senior Honors Thesis or two semesters of Senior Research may be used to satisfy part of the electives requirement, which could provide excellent research experience for students interested in pursuing a PhD.

Learning Objectives

Upon completion of this concentration, students should be able to:

- Formulate real-world problems involving data such that they can be solved by machine learning
- Implement and analyze existing learning algorithms
- Employ probability, statistics, calculus, linear algebra, and optimization in order to develop new predictive models or learning methods
- Select and apply an appropriate supervised learning algorithm for problems of different kinds, including classification, regression, structured prediction, clustering, and representation learning
- Describe the the formal properties of models and algorithms for learning and explain the practical implications of those results
- Compare and contrast different paradigms for learning

Prerequisites

The following courses are expected to be completed before the Core courses in the ML Concentration:

15-122	Principles of Imperative Computation	12
15-151	Mathematical Foundations for Computer Science	12
or 21-127	Concepts of Mathematics	
or 21-128	Mathematical Concepts and Proofs	
36-235	Probability and Statistical Inference I	9
or 36-218	Probability Theory for Computer Scientists	
or 36-219	Probability Theory and Random Processes	
or 36-225	Introduction to Probability Theory	
or 15-259	Probability and Computing	
or 21-235	Mathematical Studies Analysis I	

Course Requirements

The ML Concentration requires that students complete two core courses and their choice of three elective courses of at least 9 units each. The electives can be through a combination of coursework in Machine Learning and optionally senior research.

Core - 2 Courses:

Students must take two core courses, each being at least 9 units:

10-315	Introduction to Machine Learning (SCS Majors)	12
Plus one of:		
10-403	Deep Reinforcement Learning & Control	12
or 10-405	Machine Learning with Large Datasets (Undergraduate)	
or 10-414	Deep Learning Systems: Algorithms and Implementation	
or 10-417	Intermediate Deep Learning	
or 10-418	Machine Learning for Structured Data	
or 10-422	Foundations of Learning, Game Theory, and Their Connections	

Electives - 3 Courses:

Students need to take three courses from the following list, each being at least 9 units. Students may substitute one of these courses with one semester of an SCS Senior Honors Thesis or equivalent senior research credit.

10-403	Deep Reinforcement Learning & Control	12
or 10-703	Deep Reinforcement Learning & Control	
10-405	Machine Learning with Large Datasets (Undergraduate)	12
or 10-605	Machine Learning with Large Datasets	
10-414	Deep Learning Systems: Algorithms and Implementation	12
10-417	Intermediate Deep Learning	12
or 11-485	Introduction to Deep Learning	
or 10-707	Advanced Deep Learning	
10-418	Machine Learning for Structured Data	12
or 10-708	Probabilistic Graphical Models	
10-422	Foundations of Learning, Game Theory, and Their Connections	12
10-425	Introduction to Convex Optimization	12
or 10-725	Convex Optimization	
10-613	Machine Learning Ethics and Society	12
10-777	Historical Advances in Machine Learning	12
36-401	Modern Regression	9

Important Notes:

- To avoid excessive overlap in covered material, at most one of the core Deep Learning courses may be used to fulfill concentration course requirements: 10-417, 10-617, 11-485, 10-707. In general, students are discouraged from taking more than one of these.
- 15-281 Artificial Intelligence: Representation and Problem Solving covers several topics (i.e. reinforcement learning and Bayesian networks) that are complementary to 10-315. While not part of the ML Concentration curriculum, this course is also one to consider.
- Students should note that some of these elective courses (those at the 600-level and higher) are primarily aimed at graduate students, and so should make sure that they are adequately prepared for them before enrolling. Graduate-level cross-listings of these courses can also be used for the ML Concentration, if the student is adequately prepared for the more advanced version and the home department approves the student's registration.
- Please be aware that not all graduate-level courses in the Machine Learning Department may be used as electives. In particular, 10-606/10-607 Computational Foundations for Machine Learning may not be used as electives for the Machine Learning Concentration.

Students can apply one semester of a senior honors research thesis or research-based independent study in a topic related to this concentration, as approved by the concentration director/advisor, as one of the elective courses for this concentration. This research must have a significant communication component, including a paper or technical report, and a poster presentation. Any research course can count for at most 12 units toward the concentration and can count for at most one elective.

SCS Senior Honors Thesis

The SCS Senior Honors Thesis consists of 36 units of academic credit for this work. Up to 12 units may be counted towards the ML Concentration. Students must consult with the Computer Science Department for information about the SCS Senior Honors Thesis. Once both student and advisor agree upon a project, the student should submit a one-page research proposal to the Machine Learning Concentration Director to confirm that the project will count for the Machine Learning Concentration.

Senior Research

Senior Research consists of 2 semesters of 10-500 Senior Research Project, totaling 24 units. Up to 12 units may be counted towards the ML Concentration.

The research must be a year-long senior project, supervised or co-supervised by a Machine Learning Core Faculty member. It is almost always conducted as two semester-long projects, and must be done in senior year. Some samples of available Machine Learning Senior Projects are available on the Machine Learning Department webpage. Interested students should contact the faculty they wish to advise them to discuss the research project, before the semester in which research will take place.

Once both student and advisor agree upon a project, the student should submit a one-page research proposal to the Machine Learning Concentration Director to confirm that the project will count for the Machine Learning Concentration. The student will present the work and submit a year-end write-up to the Concentration Director at the end of Senior year.

Double Counting

At least 3 courses (each being at least 9 units) must be used for only the Machine Learning Concentration, not for any other major, minor, or concentration. (These double counting restrictions apply specifically to the Core Courses and the Electives. Prerequisites may be counted towards other majors, minors, and concentrations and do not count towards the 3 courses that must be used for only the Machine Learning Concentration.)

Advising and Management

The ML Concentration Director will hold advising office hours leading up to registration week each semester. In addition, the ML Concentration Program Coordinator will hold regular office hours to address general questions. All office hours will be detailed on the ML Concentration website.

SCS Students interested in pursuing this concentration should contact Matt Gormley for an initial advising consultation.

Principles of Programming Languages Concentration

This concentration is available to SCS students only.

Robert Harper, *Concentration Director and Advisor*
Location: GHC 9229

Amy Weis, *Concentration Coordinator*
Location: GHC 4115

Programming languages play a central role in computer science. All programs are written in a language, and it is obvious that some are better than others, at least for some purposes. The constant demand for new languages reflects the changing demands for constructing reliable and maintainable software systems. Academic research in programming language principles has led to numerous advances in language design, language implementation, and program verification intended to meet these changing expectations through the development of a rigorous theory of programming languages.

Carnegie Mellon is a recognized leader in programming languages, characterized by a strong emphasis on the centrality of type theory, a consolidation of ideas in mathematical semantics, programming logics, and programming language design and implementation. The purpose of the PoPL concentration is to teach the comprehensive view of the field that has been developed here over many decades. Type theory teaches how to define a language, and how to show that it is well-defined, free of internal contradictions. It teaches the mathematical foundations for abstraction and modularity, concepts that are fundamental to building maintainable systems. It teaches how to use a rigorous language definition as the basis for building a compiler that correctly implements the definition, and provides the tools necessary to achieve it. It teaches the logical foundations of program development, how to precisely specify the intended behavior of a program, and how to use machine tools to verify that a program meets those expectations. It gives precise meaning to language concepts, relating them to one another, and distinguishing concepts that are often confused or conflated. It teaches how to specify and verify the resource usage of a program (such as its sequential and parallel time and space complexity) without resorting to a model of how it is implemented on a machine; it supports using actual code, rather than pseudo-code, for defining and analyzing algorithms.

The PoPL concentration is of value to a broad range of students. For the practically minded it will provide the foundation for structuring and validating programs, using type systems or more advanced forms of specification. For the theoretically minded it will provide the foundation for understanding the close relationship between specification and programs on one hand and mathematical conjecture and proof on the other. The elegance of the PoPL lies in their unification of these two perspectives: the theory applies directly to the practice, and the practice informs the theory.

Learning Objectives

The PoPL concentration is characterized by a collection of learning outcomes that it seeks to achieve. These may be summarized by the knowledge that students may expect to gain by concentrating in the area. By their choice of electives each student will choose an emphasis within the area; the required courses ensure that this includes at least the first five objectives:

- Specify the concrete and abstract syntax of a programming language, including a precise specification of the binding and scope of declarations.
- Define the static semantics (compile-time constraints) of a programming language using typing judgments, and how to state and prove that it properly defined.
- Define the dynamics semantics (run-time behavior) of a language using operational and denotational methods.
- Verify rigorously that the statics and dynamics of a language are coherent, a property commonly called type safety.
- Understand the propositions-as-types principle, which relates programs to proofs and specifications to theorems, and know how to apply it in language design and program verification.
- Formulate type and assertion languages for specifying the behavior of a program, and how to verify that a program satisfies such a specification.
- Specify the cost (sequential and parallel time and space complexity) for a program written in a precisely defined language, and how to verify that a given program meets stated cost bounds.
- Use software tools to verify both the properties of languages and the specifications of programs written in well-defined languages.
- Use the static and dynamic semantics of a language to derive a compiler for it that complies with these definitions, and how to use types and verification tools to ensure compiler correctness.
- Relate a language definition to its implementation, both in terms of the run-time structures required, but also to validate abstract cost measures in an implementation.

REQUIREMENTS

This concentration requires two courses along with two additional electives.

Required Courses (complete all of the following):

15-312	Foundations of Programming Languages	12
15-317	Constructive Logic	9

Electives (complete two of the following):

15-314	Programming Language Semantics	12
15-316	Software Foundations of Security and Privacy	9
15-414	Bug Catching: Automated Program Verification	9
15-417	HOT Compilation	12
15-424	Logical Foundations of Cyber-Physical Systems	12
Any graduate-level Programming Languages course(s), with prior permission of the concentration advisor and the course instructor(s).		

Students can apply one semester of a senior honors research thesis or research-based independent study in a topic related to this concentration, as approved by the concentration director/advisor, as one of the elective courses for this concentration. This research should be supervised by a member of the Principles of Programming faculty in the Computer Science Department. This research must have a significant communication component, including a paper or technical report, and a poster presentation. Any research course can count for at most 12 units toward the concentration and can count for at most one elective.

Transfer of credit for courses taken outside of Carnegie Mellon University toward this concentration will not be allowed.

Double Counting

Either 15-312 or 15-317 (but not both) may be double counted towards any major, minor or other concentration being pursued by the student. No other double counting is permitted.

Advising and Management

Participation in this concentration is supervised by the concentration coordinator in cooperation with the students academic advisor, course instructors, and, as appropriate, thesis supervisor. The current advisor is Robert Harper. Content for this concentration will be reviewed yearly by the Principles of Programming faculty in the Computer Science Department.

Students interested in pursuing this concentration should contact Robert Harper for an initial advising consultation.

Robotics Concentration

This concentration is available to SCS students only.

Cameron Riviere, *Concentration Director/Advisor*
Location: NSH 3205

Barbara ("B.J.") Fecich, *Concentration Administrator*

The SCS Robotics Concentration provides an opportunity for SCS undergraduate students at Carnegie Mellon to learn the principles and practices of robotics through theoretical studies and hands-on experience with robots. Students initially learn the basics of robotics in an introductory robotics overview course. Additional required courses teach control systems and robotic kinematics. Students also choose from a wide selection of electives in mobile systems, machine learning, computer vision, cognition and cognitive science, or computer graphics. Students have a unique opportunity to undertake independent research projects, working under the guidance of Robotics Institute faculty members; this provides an excellent introduction to robotics practice, for those considering industry and research for those considering graduate studies.

Learning Objectives

Students completing this concentration will be able to demonstrate the following skills and learning:

- construct robots which are driven by a microcontroller through several projects, with each project reinforcing the basic principles of: vision, motion planning, mobile mechanisms, kinematics, inverse kinematics, and sensing
- apply feedback control theory to the development of robotic systems, including the principles of classical linear control theory, linear state-space methods, nonlinear systems theory, and elementary control using computer learning techniques
- program a robot arm using the principles of kinematics and dynamics: transformations, forward kinematics, inverse kinematics,

differential kinematics (Jacobians), manipulability, and the basic equations of motion

- apply related fields of computing to the construction and testing of robotic solutions: machine learning, AI, graphics and computer vision, cognitive science and learning models, cyber-physical and embedded systems
- work effectively in a team include computer and mechanical engineers to solve challenging robotics problems

Prerequisites

The following courses are expected to be completed before the Core courses in the Robotics Concentration:

CS background:		
15-122	Principles of Imperative Computation	12
15-213	Introduction to Computer Systems	12
One year of calculus:		
21-120	Differential and Integral Calculus	10
21-122	Integration and Approximation	10
Matrix Algebra (one of the following):		
21-241	Matrices and Linear Transformations	11
21-242	Matrix Theory	11
Probability (one of the following):		
15-259	Probability and Computing	12
21-325	Probability	9
36-218	Probability Theory for Computer Scientists	9
36-225	Introduction to Probability Theory	9

The probability course can be taken concurrently with the concentration requirements. Depending on specific electives chosen, additional prerequisites may be required (e.g. 21-259).

Course Requirements

The Robotics Concentration requires that students complete three core courses and their choice of two elective courses of at least 9 units each. The electives can be chosen from a specific set of stand-alone courses. Students can opt to do an undergraduate research project as one of their electives.

Required core courses (36 units)

16-311	Introduction to Robotics	12
plus the following two courses:		
16-299	Introduction to Feedback Control Systems	12
16-384	Robot Kinematics and Dynamics	12

Electives (minimum 18 units)

Students must complete 2 electives from the following list of courses for a minimum of 18 units. At least one of the two electives courses must be from the Robotics Institute (16-xxx). A maximum of 12 units of research (16-597) can be used toward this requirement.

16-264	Humanoids	12
16-362	Mobile Robot Algorithms Laboratory	12
16-385	Computer Vision	12
16-421	Vision Sensors	12
16-423	Designing Computer Vision Apps	12
16-597	Undergraduate Reading and Research	Var.
10-315	Introduction to Machine Learning (SCS Majors)	12
11-344	Machine Learning in Practice	12
15-281	Artificial Intelligence: Representation and Problem Solving	12
15-348	Embedded Systems	9
15-424	Logical Foundations of Cyber-Physical Systems	12
15-462	Computer Graphics	12
15-463	Computational Photography	12
15-494	Cognitive Robotics: The Future of Robot Toys	12
85-370	Perception	9
85-395	Applications of Cognitive Science	9
85-412	Cognitive Modeling	9
85-419	Introduction to Parallel Distributed Processing	9
85-426	Learning in Humans and Machines	9

Note: Graduate level Robotics courses may be used to meet the elective requirement with permission from the Concentration Advisor. Graduate level Mechanical Engineering and Electrical and Computer Engineering courses that are relevant to robotics may be used to meet the elective requirement with permission from the Concentration Advisor.

Students can apply one semester of a senior honors research thesis or research-based independent study in a topic related to this concentration, as approved by the concentration director/advisor, as one of the elective courses for this concentration. This research must have a significant communication component, including a paper or technical report, and a poster presentation. Any research course can count for at most 12 units toward the concentration and can count for at most one elective.

Double Counting

Courses being used to satisfy the requirements for the Robotics Concentration may not be counted towards another minor or concentration. A minimum of 27 units must be completed that is not double counted with any other program. This allows a maximum of 2 courses that can double count with other programs.

Advising and Management

The Robotics Concentration will always have a Robotics faculty member as its Advisor. There will also be a Robotics Concentration Administrator who will track progress for any SCS student pursuing this concentration. Curriculum changes will be discussed with the Robotics Institute faculty and SCS academic deans.

Students interested in this concentration should set up an initial advising consultation with the Concentration Advisor to assess interest and preparation.

Security & Privacy Concentration (SCS)

This concentration is available to SCS students only.
(ECE students should consult their department for an equivalent concentration.)

Lujo Bauer, *Concentration Director and Advisor*
Location: CIC 2203

In a world where data breaches and cyber-attacks are ever-present, the need for technologists who have a solid understanding of the principles that underlie strong security and privacy practices is greater than ever.

The Security & Privacy concentration is designed to expose students to the key facets of and concerns about computer security and privacy that drive practice, research, and legislation. On completing the curriculum, students will be well prepared to continue developing their interests in security or privacy through graduate study; to take jobs in security or privacy that will provide further training in applicable areas; and to be informed participants in public and other processes that shape how organizations and society develop to meet new challenges related to computer security or privacy.

Learning Objectives

After completing this concentration, students should:

- Understand how to reason about the adversary in computer systems.
- Be familiar with common security vulnerabilities, from buffer overflows and return oriented programming to cross-site scripting, and widely deployed defenses against these vulnerabilities.
- Be familiar with and understand how to apply the basic concepts in cryptography and secure system design and analysis.
- Understand the key properties of commonly used cryptographic primitives and properties commonly desired of cryptographic protocols.
- Be familiar with current and upcoming research directions in secure system design, software analysis, and cryptography.
- Be familiar with the breadth of concerns and topics relevant to computer security and privacy, ranging from technical topics to ethics, regulation, usability, and economics.
- Be familiar with the key concepts in privacy, ranging from conceptions of privacy to privacy algorithms to regulation and policy.
- Gain a more in-depth understanding of one “context” area: usable security and privacy, or policy.

Prerequisites

Students interested in pursuing this concentration should have the following courses completed before starting the concentration:

15-151	Mathematical Foundations for Computer Science	12
15-213	Introduction to Computer Systems	12
15-251	Great Ideas in Theoretical Computer Science	12

Curriculum

A distinguishing feature of this field is the ubiquitous need to consider an adversary, and the resulting interplay between attack and defense that routinely advances both theory and practice. In order to understand widely-deployed defensive techniques and secure-by-design approaches, students must also understand the attacks that motivate them and the “adversarial mindset” that leads to new forms of attack. The curriculum is designed around this principle.

Students in the Security & Privacy concentration will take courses that cover the basic principles *(Introduction and Basics Course Area)*, the underlying theory *(Theoretical Foundations Course Area)*, and the practical application *(System Design Course Area)* of security and privacy. Additionally, they will be required to select a course which covers either usability or policy *(Context Course Area)*. Finally, students will have the opportunity to dive deep on a particular security & privacy topic by completing an elective of their choosing *(Depth Course Area)*.

Requirements (5 courses, minimum 48 units):

Introduction/Portal Entry course		Units
15-330	Introduction to Computer Security	12

Crosslisted courses allowed.

Theoretical Foundations course (choose one option):		Units
15-356	Introduction to Cryptography	12
or both:		
17-731	Foundations of Privacy	12
18-733	Applied Cryptography	

Crosslisted courses allowed.

System Design course (choose one):		Units
15-316	Software Foundations of Security and Privacy	9
18-335	Secure Software Systems	12

Context Course (select one):		Units
17-334	Usable Privacy and Security	9
or one of:		
17-333	Privacy Policy, Law, and Technology	9
17-731	Foundations of Privacy (not available if used for Theoretical Foundations requirement)	12

Depth Course (complete one option below):		Units
Option 1: Complete an elective course from the below list of approved electives, at least 9 units		9
Option 2: Complete an additional course from the prior categories		9
Option 3: Complete at least 9 units of Independent Study - this option must be pre-approved by your primary advisor as well as the Concentration Director, and an appropriate faculty advisor and study plan submitted.		9
Approved Elective List:		
11-488	Concepts in Digital Multimedia and Cyber Forensics	12
14-795	AI Applications in Information Security	
14-828	Browser Security	
14-809	Introduction to Cyber Intelligence	
14-814	Wireless Network Security	
14-819	Introduction to Software Reverse-Engineering	
14-822	Host-Based Forensics	
14-823	Network Forensics	
14-829	Mobile and IoT Security	
15-435	Foundations of Blockchains	12
17-303	Cryptocurrencies, Blockchains and Applications	9
17-735	Engineering Privacy in Software	12
17-880	Algorithms for Private Data Analysis	12
18-334	Network Security	12
18-632	Introduction to Hardware Security	
18-636	Browser Security	

Crosslisted courses allowed.

Students can apply one semester of a senior honors research thesis or research-based independent study in a topic related to this concentration, as approved by the concentration director/advisor, as one of the elective courses for this concentration (see option 3 above). This research must have a significant communication component, including a paper or technical report, and a poster presentation. Any research course can count for at most 12 units toward the concentration and can count for at most one elective.

Anti-requisites

When two (or more) courses overlap significantly in the material they cover, only one can count toward the security and privacy concentration. An example pair is 15-316 Software Foundations of Security and Privacy and 18-335 Secure Software Systems. Another excluded pair is 15-356 Introduction to Cryptography and 18-733 Applied Cryptography. Other such anti-requisites may occur; please consult the concentration director when scheduling courses.

Excluded Courses

Some security and privacy courses may not be counted towards concentration requirements. These courses all serve specific important different purposes, but do not fit into the concentration as currently designed. For example, 17-331 is more suitable for students who are interested in a broader single-course introduction to information security, but has too much overlap with the concentration’s required intro course to be able to count toward the concentration. For other excluded courses, please consult the concentration website.

Double Counting

Only two of the courses that are counted toward concentration requirements can also be counted towards core course requirements of other majors and minors being pursued by the student. SCS or ECE General Education Requirements are not included in this double-counting limit.

Advising and Management

The concentration is open to all undergraduates in the School of Computer Science. There is no formal admissions process. Students intending to pursue the concentration should contact the concentration coordinator to register their intention. Curriculum for the concentration will be reviewed by the coordinator annually in consultation with faculty in supporting departments.

Software Engineering Concentration

This concentration is available to SCS students only.

Michael Hilton, *Concentration Director and Advisor*
Location: Wean Hall 5122

Effectively building modern software systems at scale requires not just programming skills, but also engineering skills. These skills include the ability to interact effectively with customers to gather the requirements for a system in a precise way; to develop a design that resolves competing quality attributes; to make tradeoffs among schedule, cost, features, and quality to maximize value to stakeholders; to work effectively with other engineers; and to assure the quality of the delivered software system. We hear regularly from industry that these skills are crucial to them, and that they are interested in students with a strong software engineering background.

Learning Objectives

The software engineering concentration is designed to teach the fundamental tools, techniques, and processes of software engineering. Through internships and a mentored project experience, students gain an understanding of the issues of scale and complexity that motivate software engineering tools and techniques. The core curriculum includes material both on engineering the software product and on the process, teamwork, and management skills that are essential to successful engineering. Graduates of the program should have the technical, process, and teamwork skills to be immediately productive in a mature engineering organization.

Prerequisites

SCS students should have completed the following courses before completing this concentration:

15-122	Principles of Imperative Computation	12
15-151	Mathematical Foundations for Computer Science	12
15-213	Introduction to Computer Systems	12
17-214	Principles of Software Construction: Objects, Design, and Concurrency (This prerequisite exists to ensure software development experience, and may be waived if the student can show significant development experience. Please contact the Concentration Director to request this prerequisite be waived.)	12

Course Requirements

Overall, the concentration requires five (5) courses and an internship. One of the five courses consists of a 6-credit course that serves as a writing workshop in which the student reflects upon and integrates the lessons of the internship.

Required Courses (complete all of the following):

17-313	Foundations of Software Engineering	12
17-413	Software Engineering Practicum	12

Electives (complete each category):

Domain-Independent Elective course or mini sequence focused on technical software engineering material (minimum 9 units required):

At least 9 units total.		
15-316	Software Foundations of Security and Privacy	9
15-414	Bug Catching: Automated Program Verification	9
17-355	Program Analysis	12
17-356	Software Engineering for Startups	12
17-731	Foundations of Privacy	12
17-821	Computer Simulation of Complex Socio-Technical Systems	12
Or combine the following Mini-Sequences:		
17-314	Formal Methods *Combined with 17-324	6
17-324	Advanced Formal Methods *	6
17-612	Business and Marketing Strategy **Combined with either 17-626 or 17-627	6
17-626	Requirements for Information Systems **	6
17-627	Requirements for Embedded Systems **	6
17-322	Agile Methods ***Combined with 17-332 and 17-442	6
17-332	Software Project Management ***	6
17-442	Software Management Theory ***	6
17-323	Quality Assurance ****Combined with 17-443	6
17-443	Quality Management ****	6
17-335	Software Architectures May be combined additionally with any of the above to achieve 9+ units total.	6

Crosslisted courses allowed.

Additional courses may count with prior approval by the concentration director.

Computer Science in Society & Industry Elective: a course addressing problems related to existing and emerging technologies and their associated social, political, legal, business, and organizational contexts (minimum 9 units required):

17-200	Ethics and Policy Issues in Computing	9
17-331	Information Security, Privacy, and Policy	12
17-333	Privacy Policy, Law, and Technology	9
17-334	Usable Privacy and Security	9
17-562	Law of Computer Technology	9
19-403	Policies of Wireless Systems	12
70-311	Organizational Behavior	9
70-415	Introduction to Entrepreneurship	9
70-471	Supply Chain Management	9

Crosslisted courses allowed.

See concentration website for additional course options; additional courses may be counted if prior-approval is given by the Concentration Director.

Students can apply one semester of a senior honors research thesis or research-based independent study in a topic related to this concentration, as approved by the concentration director/advisor, as one of the elective courses for this concentration. This research must have a significant communication component, including a paper or technical report, and a poster presentation. Any research course can count for at most 12 units toward the concentration and can count for at most one elective. Consult with the concentration director on what elective can be replaced in this option.

Internship and Reflection

The concentration requires a software engineering internship of a minimum of 8 full-time weeks in an industrial setting (i.e., integrated into a team and exposed to industry pressures). The student may work in development, management, quality assurance, or other relevant positions. Students should confirm with the director that an internship is appropriate, but internships that fulfill the criteria will be accepted after the fact. Students must further complete:

17-415	Software Engineering Reflection	6

Double Counting

No more than two of the courses used to fulfill the concentration requirements may be counted towards any other degree or concentration. SCS General Education requirements do not count towards this restriction.

Advising and Management

The concentration coordinator is responsible for academic advising, handling exceptions and updating the curriculum each year, in consultation with faculty in the Institute for Software Research.

Students who are interested in pursuing this concentration should contact the concentration coordinator for an initial advising consultation.