Additional Majors and Minors in SCS

This page lists Additional Majors and Minors apart from those in Artificial Intelligence, Computational Biology (http://coursecatalog.web.cmu.edu/schoolofcomputerscience/undergraduatecomputatonalbiology) and Computer Science (http://coursecatalog.web.cmu.edu/schoolofcomputerscience/undergraduatecomputerscience). Click on a tab to see more information about each program.

Students should consult with their own academic advisor as well as the advisor for the given minor for specific double-counting rules, especially for students who are pursuing an SCS minor with a major or other minors closely related to computing. Additional help can be provided by the Assistant Dean in the Computer Science Undergraduate Program office (Gates-Hillman Center 4th Floor).

SCS Concentrations: SCS concentrations will begin to launch during the 2018-19 academic year. Consult with your academic advisor during your first academic year for new SCS concentrations that will launch. SCS concentrations will be open to all SCS students. For CS Majors entering CMU in 2018 or later, these students will be required to pursue a minor outside of SCS or a concentration within SCS; minors in SCS will not be allowed for these students. Computer Science majors entering prior to 2018 can substitute an SCS concentration for the minor requirement if available and approved by their academic advisor and program director. Artificial Intelligence and Computational Biology majors can complete an SCS concentration if they wish, but it is not required for these degrees. Consult the SCS undergraduate programs website (https://www.cs.cmu.edu/undergraduate-programs) for information about these concentrations as they are approved.

Human-Computer Interaction Additional Major

Vincent Aleven, Undergraduate Director
Office: Newell Simon Hall (NSH) 3531
For up to date information, see: http://www.hcii.cmu.edu/

Overview

Human-Computer Interaction (HCI) is a fast growing field devoted to the design, implementation, and evaluation of interactive computer-based technology. Examples of HCI products include intelligent computer tutors, wearable computers, social networking sites, and internet connected personal digital assistants (PDAs). Constructing an HCI product is a cyclic, iterative process that has at least three stages: Design, Implementation, and Evaluation.

The Design stage involves principles of design and human behavior, the Implementation stage principles of computer science, and the Evaluation stage empirical research methods common to several disciplines. There are thus four topical areas to cover in this major: Human Behavior, Design, Implementation, and Evaluation. In slightly more detail, the major involves the following sorts of knowledge and skill:

Design

- Eliciting from the client, formulating, and articulating functional specifications
- Knowing how human factors and cognitive models should inform design
- Knowing the principles of, and having experience with, communication design
- Understanding how implementation constraints should inform design
- Incorporating evaluation results into iterated designs

Implementation Programming Skills

- Standard programming languages - e.g., C++, Java
- Rapid prototyping skill (e.g., Visual Basic, Flash)
- Computational literacy, i.e., knowledge sufficient for effective communication and decision making about:
- interface construction tools and languages
- multimedia authoring tools
- data structures and algorithms
- Operating systems, platforms, etc.

Evaluation

- Experimental design
- Focus Groups
- Surveys
- Usability Testing (Cognitive walkthroughs, user models, heuristic evaluation, GOMS)
- Statistical Analysis

There are over 45 courses relevant to these areas that are now offered by eight different departments in four different colleges at Carnegie Mellon (the School of Computer Science, the Dietrich College of Humanities and Social Sciences, and the College of Fine Arts, and the Tepper School of Business).

Curriculum

Required Courses

Cognitive Psychology		Units
85-211	Cognitive Psychology	9
or 85-213	Human Information Processing and Artifical Intelligence	
Interaction Design Studio 1		
05-651	Interaction Design Studio 1 [c]	12
or 51-261	Design Center: Communication Design Fundmntls: IxD for Communications	
or 51-262	Communication Design Fundamentals: Design for Interactions for Communications	
Statistics (one of the following):		
36-200	Reasoning with Data	9
36-207	Probability and Statistics for Business Applications	9
36-220	Engineering Statistics and Quality Control	9
36-225-36-226	Introduction to Probability Theory - Introduction to Statistical Inference [b]	18
36-247	Statistics for Lab Sciences	9
70-207	Probability and Statistics for Business Applications	9
Introduction to Programming		
15-104	Introduction to Computing for Creative Practice	10
or 15-110	Principles of Computing	
or 15-112	Fundamentals of Programming and Computer Science	
or 15-121	Introduction to Data Structures	
Interaction Design Studio 2		
05-650	Interaction Design Studio II	12
Human-Computer Interaction Methods		
05-410	User-Centered Research and Evaluation	12
Interface Programming		
05-430	Programming Usable Interfaces [a]	15
Project Course		
05-571	Undergraduate Project in HCI	12

Notes

- [a] The required HCI programming course 05-430 Programming Usable Interfaces is only guaranteed to be offered in the Fall. Spring offerings are only when instructor resources are available. When you register for this course, you must also sign up for a recitation time, which is equivalent to the User Interface Lab. The labs differ on their computer science prerequisites. Section D should be taken by students majoring in computer science or with advanced technical skills. Section A through C require only an introductory course in computer science as a prerequisite, and can be taken either by computer science majors or non-computer science majors.
- [b] The statistics course is required so that majors will be able to understand and conduct empirical research in HCI. Therefore a mathematically-oriented probability course, such as 36-217 Probability Theory and Random Processes does not fulfill either requirement. However, the sequence of 36-225 Introduction to Probability Theory and 36-226 Introduction to Statistical Inference (i.e., a mathematical statistics course followed by a statistical inference course) fulfills the statistics prerequisite requirement.
- [c] Design majors do not need to take 05-651 Interaction Design Studio 1 as a prerequisite, since they learn similar material in other courses for their major. 51-262 Communication Design Fundamentals: Design for Interactions for Communications and 51-261 Design Center: Communication Design Fundmntls: IxD for Communications also count as fulfilling this requirement.

a The required HCI programming course 05-430 Programming Usable Interfaces is only guaranteed to be offered in the Fall. Spring offerings are only when instructor resources are available. When you register for this course, you must also sign up for a recitation time, which is equivalent to the User Interface Lab. The labs differ on their computer science prerequisites. Section D should be taken by students majoring in computer science or with advanced technical skills. Section A through C require only an introductory course in computer science as a prerequisite, and can be taken either by computer science majors or non-computer science majors.

b The statistics course is required so that majors will be able to understand and conduct empirical research in HCI. Therefore a mathematically-oriented probability course, such as 36-217 Probability Theory and Random Processes does not fulfill either requirement. However, the sequence of 36-225 Introduction to Probability Theory and 36-226 Introduction to Statistical Inference (i.e., a mathematical statistics course followed by a statistical inference course) fulfills the statistics prerequisite requirement.

c Design majors do not need to take 05-651 Interaction Design Studio 1 as a prerequisite, since they learn similar material in other courses for their major. 51-262 Communication Design Fundamentals: Design for Interactions for Communications and 51-261 Design Center: Communication Design Fundmntls: IxD for Communications also count as fulfilling this requirement.

Electives (18 Units)

Electives are intended to provide HCI double majors advanced concepts and skills relevant to HCI or breadth of experience not available from their primary major. Given these goals, most electives will be 300-level courses or higher. Courses at the 100-level and 200-level in one's primary major will

not count as electives, although the same course taken by a non-major may count (approval is still required).

Students can take electives in the HCII or courses relevant to HCI from many other departments on campus. All external electives are approved on a case-by-case basis.

The following courses have been approved as electives in the past, organized by the offering department:

Human-Computer Interaction		Units
05-291	Learning Media Design	12
05-320	Social Web	12
05-395	Applications of Cognitive Science	9
05-413	Human Factors	9
05-418	Design Educational Games	12
05-432	Personalized Online Learning	12
05-434	Machine Learning in Practice	12
05-452	Service Design	12
05-499	Special Topics in HCI	12
05-540	Rapid Prototyping of Computer Systems	12
05-589	Independent Study in HCI-UG	Var.
05-823	E-Learning Design Principles and Methods	12
Machine Learning		
10-601	Introduction to Machine Learning (Masters)	12
Computer Science		
15-390	Entrepreneurship for Computer Science	9
15-421	Information Security and Privacy	12
15-437	Web Application Development	12
15-462	Computer Graphics	12
15-466	Computer Game Programming	12
Statistics		
36-309	Experimental Design for Behavioral & Social Sciences	9
Architecture		
48-339	IDeATe: Making Things Interactive	12
Design		
51-241	How People Work	9
51-324	Basic 3D Prototyping	4.5
51-327	Design Center: Introduction to Web Design	9
51-328	Advanced Web Design	9
51-383	Topics: Conceptual Models	9
51-385	Design for Service	9
51-424	Web Portfolio	4.5
51-359	Tools for UX Design	9
Business Administration		
70-414	Entrepreneurship for Engineers	9

Double Counting

All prerequisites can be double counted with any requirements in your primary major. At most, two non-prerequisite courses can be double counted with core requirements in primary majors.

Accelerated Master's Programs

The HCII currently offers a three semester (12-month), 15 course Masters in HCI. Undergraduates currently enrolled in the HCI major may apply for the Accelerated Masters program in the fall semester of their senior year. If admitted, student finish the masters degree the following Fall semester.

Admission to the Major

The HCI undergraduate major is currently available only as a additional major. Because space is limited in the major's required courses, enrollment in the HCI undergraduate major is currently limited to about 35 students in each graduating class. The admissions period occurs in spring semesters. For more details, see the website: https://hcii.cmu.edu/academics/hci-undergraduate.

Human-Computer Interaction Minor

The Minor in Human-Computer Interaction will give students core knowledge about techniques for building successful user interfaces, approaches for conceiving, refining, and evaluating interfaces that are useful and useable, and techniques for identifying opportunities for computational technology to improve the quality of people's lives. The students will be able to effectively collaborate in the design, implementation, and evaluation of easy-to-use, desirable, and thoughtful interactive systems. They will be prepared to contribute to multidisciplinary teams that create new interactive products, services, environments, and systems.

The key concepts, skills and methods that students will learn in the HCI Minor include:

- Fieldwork for understanding people's needs and the influence of context
- Generative approaches to imagining many possible solutions such as sketching and "bodystorming"
- Iterative refinement of designs
- Basic visual design including typography, grids, color, and the use of images
- Implementation of interactive prototypes
- Evaluation techniques including discount and empirical evaluation methods

The HCI minor is targeted at undergraduates who expect to get jobs where they design and/or implement information technology-based systems for end users, and well as students with an interest in learning more about the design of socio-technical systems. It is appropriate for students with majors in Computer Science and Information Systems, as well as students in less software-focused majors, including Design, Architecture, Art, Business Administration, Psychology, Statistics, Decision Science, Mechanical Engineering, Electrical Engineering, English and many others in the university.

Curriculum

The only prerequisite for this Minor is an introductory-level college programming course (such as 15-110, 15-112, 15-121, or 51-257) and to be in good standing with the University.

In addition to the programming prerequisite, the Minor has required two courses—05-391 Designing Human Centered Software (DHCS) and 05-392 Interaction Design Overview (IxDO)—and four electives. The student will be required to get a grade of "C" or better in each course in order for it to count as part of the Minor. There is no final project or research required for the Minor.

Required Courses

- 05-391 Designing Human Centered Software (DHCS)[1]: This course provides an overview of the most important methods taught in the Additional Major in HCI, such as Contextual Inquiry, Prototyping and Iterative Design, Heuristic Evaluation, and Think Aloud User Studies. It covers in a more abbreviated form the content of 05-410 User-Centered Research and Evaluation, 05-430 Programming Usable Interfaces.
- 05-392 (IxDO)[2]: This is a design course that will combine material from 05-651 *and* 05-650 for students who do not have any previous experience with design, in a form that will fit appropriately in to a one-semester format.

Electives

The HCI minor requires four electives approved by the undergraduate director.

Double Counting

Students may double count up to two (2) of the required courses or electives with any other major or minor.

Relationship between the BHCI Major and Minor

Admission

- **BHCI Major:** Application and admissions required, information on the HCII website (http://www.hcii.cmu.edu).
- **BHCI Minor:** Admissions form available at the HCII website (http://www.hcii.cmu.edu).

Prerequisites

- **BHCI Major:**
  - Freshman-level programming
  - Statistics
  - Cognitive Psychology
  - Interaction Design Studios
- **BHCI Minor:**
  - Freshman-level programming

Core Courses

- **BHCI Major:**
  - Interaction Design Studio I & II (IxDS)
  - User Centered Research & Evaluation (UCRE)
  - Interface Programming (PUI)
  - BHCI Project
- **BHCI Minor:**
  - Interaction Design Overview (IxDO)
  - Designing Human Centered Systems (DHCS)

Electives

- **BHCI Major:** Four (4) electives
- **BHCI Minor:** Four (4) electives

Double Counting

- **BHCI Major:** Two (2) core courses or electives with primary major.
- **BHCI Minor:** Two (2) core courses or electives with primary major.

Footnotes

1 Alternatively, a student can take *both* the BS/MHCI empirical methods course (05-410) *and* the BS/MHCI core-programming course (either 05-430 Programming Usable Interfaces . If students take this course sequence, they would get credit for fulfilling this requirement plus one elective.

2 Alternatively, students can fulfill the design requirement by taking both 05-650 *and* 05-651. If students take this course sequence, they would get credit for fulfilling this requirement plus one elective.

These alternative ways of fulfilling the requirements for the HCI minor are designed for students who are in the HCI 2nd major who want to "downgrade" to the minor. These students can use some the courses completed for the HCI 2nd major as a way of fulfilling the requirements for the minor.

Students who are in the HCI minor right from the start are strongly encouraged to follow the regular requirements outlined above and are strongly discouraged from trying these alternative ways of fulfilling the requirements. It can be extremely difficult to get into any of the alternative courses. This is true especially for 05-650, but for other courses as well. The fact that a student in the minor has already taken 05-651 will not give priority for getting into 05-650.

IDeATe Minors

Advisor: Kelly Delaney
E-mail: kellydel@andrew.cmu.edu
Website: https://ideate.cmu.edu

The Integrative Design, Arts and Technology (IDeATe) network offers students the opportunity to become immersed in a collaborative community of faculty and peers who share expertise, experience, and passions at the intersection of arts and technology. Students engage in active "learning by doing" in state-of-the-art maker spaces. The program addresses current and emerging real-world challenges that require disciplinary expertise coupled with multidisciplinary perspectives and collaborative integrative approaches.

The IDeATe undergraduate curriculum consists of eight areas, all of which can be taken as minors. The themes of these areas integrate knowledge in technology and the arts. Four of these minors are based in the School of Computer Science:

Animation & Special Effects Minor

Explore the technical and artistic aspects of 3D and 2D animation in an integrated manner and within different application contexts, from film animation and special effects to interactive displays.

Curriculum

One Computing Course - Minimum of 9 Units

		Units
15-104	Introduction to Computing for Creative Practice	10
15-110	Principles of Computing	10
15-112	Fundamentals of Programming and Computer Science	12
60-210	Electronic Media Studio: Introduction to Interactivity	10
60-212	Electronic Media Studio: Interactivity and Computation for Creative Practice	12

One IDeATe Portal Course - Minimum of 9 Units

		Units
16-223	IDeATe Portal: Creative Kinetic Systems	10
18-090	Twisted Signals: Multimedia Processing for the Arts	10
60-223	IDeATe: Introduction to Physical Computing	10
62-150	IDeATe Portal: Introduction to Media Synthesis and Analysis	10
99-361	IDeATe Portal	9

IDeATe Animation & Special Effects Courses - Minimum of 27 Units

		Units
15-365/60-422	Experimental Animation	12
15-463	Computational Photography	12
15-465/60-414	Animation Art and Technology	12
16-374/60-428	IDeATe: Art of Robotic Special Effects	12
60-125	IDeATe Introduction to 3D Animation	12
60-220	IDeATe Technical Character Animation	10
60-333	IDeATe: Character Rigging for Production	10
60-398	Social History of Animation	9
60-410	Advanced ETB: Moving Image Magic: Visual Effects and Motion Graphics	10
60-415	Advanced ETB: Animation Studio	10
60-417	Advanced ETB: Video	10
76-285	Team Communication	6

Double-Counting

Students may double-count up to two of their Animation & Special Effects minor courses toward other requirements.

Intelligent Environments Minor

Develop spaces and devices that support efficiency and high quality of experience in contexts like daily activity, built environment, making process (from laying plaster to robot development), and arts performance.

Curriculum

One Computing Course - Minimum of 9 Units

		Units
15-104	Introduction to Computing for Creative Practice	10
15-110	Principles of Computing	10
15-112	Fundamentals of Programming and Computer Science	12
60-210	Electronic Media Studio: Introduction to Interactivity	10
60-212	Electronic Media Studio: Interactivity and Computation for Creative Practice	12

One IDeATe Portal Course - Minimum of 9 Units

		Units
16-223	IDeATe Portal: Creative Kinetic Systems	10
18-090	Twisted Signals: Multimedia Processing for the Arts	10
60-223	IDeATe: Introduction to Physical Computing	10

62-150	IDeATe Portal: Introduction to Media Synthesis and Analysis	10
99-361	IDeATe Portal	9

IDeATe Intelligent Environments Courses - Minimum of 27 Units

		Units
16/54-375	IDeATe: Robotics for Creative Practice	10
16-455/48-530	IDeATe: Human-Machine Virtuosity	12
16-467	Human Robot Interaction	12
18/05-540	Rapid Prototyping of Computer Systems	12
48/53-558	Reality Computing	12
54/16-371	Personalized Responsive Environments	9
62-315	InterBreeding Architecture: Computational Techniques for Shaping the Environment	9
76-285	Team Communication	6

Double-Counting

Students may double-count up to two of their Intelligent Environments minor courses toward other majors and minors.

Design for Learning Minor

Design effective new media systems for learning using new technologies, learning science principles and media arts knowledge. Produce engaging and effective experiences from games to tangible learning tool kits and remote systems.

Curriculum

One Computing Course - Minimum of 9 Units

		Units
15-104	Introduction to Computing for Creative Practice	10
15-110	Principles of Computing	10
15-112	Fundamentals of Programming and Computer Science	12
60-210	Electronic Media Studio: Introduction to Interactivity	10
60-212	Electronic Media Studio: Interactivity and Computation for Creative Practice	12

One IDeATe Portal Course - Minimum of 9 Units

		Units
16-223	IDeATe Portal: Creative Kinetic Systems	10
18-090	Twisted Signals: Multimedia Processing for the Arts	10
60-223	IDeATe: Introduction to Physical Computing	10
62-150	IDeATe Portal: Introduction to Media Synthesis and Analysis	10
99-361	IDeATe Portal	9

IDeATe Design for Learning Courses - Minimum of 27 Units

05-291	Learning Media Design	12
05-292	Learning Media Methods	12
05-418	Design Educational Games	12
05-432	Personalized Online Learning	12
05-823	E-Learning Design Principles and Methods	12
51-486	Learner Experience Design	9
76-285	Team Communication	6
80-292	Learning Science Principles	12
85-392	Human Expertise	9

Double-Counting

Students may double-count up to two of their Design for Learning minor courses toward requirements for other majors and minors.

Physical Computing Minor

Build interfaces and circuitry to embed in physical contexts, such as mobile environments and new creative practice instruments.

Curriculum

One Computing Course - Minimum of 9 Units

		Units
15-104	Introduction to Computing for Creative Practice	10
15-110	Principles of Computing	10
15-112	Fundamentals of Programming and Computer Science	12
60-210	Electronic Media Studio: Introduction to Interactivity	10
60-212	Electronic Media Studio: Interactivity and Computation for Creative Practice	12

One IDeATe Portal Course - Minimum of 9 Units

		Units
16-223	IDeATe Portal: Creative Kinetic Systems	10
18-090	Twisted Signals: Multimedia Processing for the Arts	10
60-223	IDeATe: Introduction to Physical Computing	10
62-150	IDeATe Portal: Introduction to Media Synthesis and Analysis	10
99-361	IDeATe Portal	9

IDeATe Physical Computing Courses - Minimum of 27 Units

		Units
15-294	Special Topic: Rapid Prototyping Technologies	5
16-374/60-428	IDeATe: Art of Robotic Special Effects	12
16/54-375	IDeATe: Robotics for Creative Practice	10
16-455/48-530	IDeATe: Human-Machine Virtuosity	12
18/05-540	Rapid Prototyping of Computer Systems	12
18-578	Mechatronic Design	12
24-672	Special Topics in DIY Design and Fabrication	12
39-245	Rapid Prototype Design	9
48-339	IDeATe: Making Things Interactive	12
48-390	Physical Computing Studio	10
48/53-558	Reality Computing	12
48-739	Making Things Interactive (Graduate)	12
60-412	Interactive Art and Computational Design	12
62-478	IDeATe: digiTOOL	6
76-285	Team Communication	6

Double-Counting

Students may double-count up to two of their Physical Computing minor courses toward requirements for other majors and minors.

Language Technologies Minor

Chair: Alan W. Black
E-mail: awb@cs.cmu.edu
Website: http://www.lti.cs.cmu.edu/learn

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 minor that gives you the opportunity to not only learn about language technologies, but to also apply that knowledge through a directed project.

Prerequisites

Prerequisites		Units
15-122	Principles of Imperative Computation	10
15-150	Principles of Functional Programming	10
Recommended		
21-241	Matrices and Linear Transformations	10
or 21-242	Matrix Theory	
36-218	Probability Theory for Computer Scientists	9

or 15-259	Probability and Computing	
or 15-359	Probability and Computing	

Curriculum

Core Course		
11-421	GRAMMARS & LEXICONS	12
or 11-721	Grammars and Lexicons	
Electives (choose 3)		
11-411	Natural Language Processing	12
11-441	Machine Learning for Text Mining	9
11-442	Search Engines	9
11-492	Speech Processing	12
11-711	Algorithms for NLP	12
11-731	Machine Translation and Sequence-to-Sequence Models	12
11-751	Speech Recognition and Understanding	12
11-752	Speech II: Phonetics, Prosody, Perception and Synthesis	12
11-761	Language and Statistics	12
80-180	Nature of Language	9
80-280	Linguistic Analysis	9
Project		
A semester-long directed research project OR paper to provide hands-on experience and an in-depth study of a topic (in same area as a chosen elective)		12

Double Counting of Courses

Students may double count 11-421/11-721 Grammars and Lexicons as well as 80-180 Nature of Language toward any other major or minor.

Machine Learning Minor

Program Director: Dr. Matt Gormley
Program Coordinator: Dorothy Holland-Minkley
E-mail: ml-minor@cs.cmu.edu
Website: http://www.ml.cmu.edu/academics/minor-in-machine-learning.html

Machine learning and statistical methods are increasingly used in many application areas including natural language processing, speech, vision, robotics, and computational biology. The Minor in Machine Learning allows undergraduates to learn about the core principles of this field.

Prerequisites

The 5 prerequisite courses must be taken before a student applies to the Machine Learning Minor.

Prerequisites		Units
15-122	Principles of Imperative Computation	10
21-120	Differential and Integral Calculus	10
21-122	Integration and Approximation	10
36-217	Probability Theory and Random Processes	9
or 36-218	Probability Theory for Computer Scientists	
or 36-225	Introduction to Probability Theory	
or 15-259	Probability and Computing	
or 15-359	Probability and Computing	
or 21-325	Probability	
36-226	Introduction to Statistical Inference	9
or 36-326	Mathematical Statistics (Honors)	

Core Courses

The Machine Learning Minor has 2 core courses taken by all students.

Core Courses		
10-401	Introduction to Machine Learning (Undergrad)	12
or 10-601	Introduction to Machine Learning (Masters)	
36-401	Modern Regression	9

Electives

The Machine Learning Minor requires at least 3 electives of at least 9 units each in Machine Learning. This can be through a combination of stand-alone courses in Machine Learning, senior research (taken over two semesters and counting as two electives), and a variety of two-course sequences that provide depth in different areas.

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 Minor, if the student is adequately prepared for the more advanced version and the home department approves the student's registration.

Stand-Alone Electives

Students can take as many of these courses as desired, with each one counting as one elective.

Take as many of these courses as desired:		
10-405	Machine Learning with Large Datasets (Undergraduate)	12
or 10-605/805	Machine Learning with Large Datasets	
10-701	Introduction to Machine Learning (PhD)	12
10-702	Statistical Machine Learning	12
10-703	Deep Reinforcement Learning & Control	12
10-707	Topics in Deep Learning	12
11-777	Advanced Multimodal Machine Learning	12
36-315	Statistical Graphics and Visualization	9
36-402	Advanced Methods for Data Analysis	9
36-461	Special Topics: Statistical Methods in Epidemiology	9
36-462	Special Topics: Data Mining	9
36-463	Special Topics: Multilevel and Hierarchical Models	9
36-464	Special Topics: Applied Multivariate Methods	9
36-700	Probability and Mathematical Statistics	12
or 36-705	Intermediate Statistics	

Senior Research

Senior research consists of 2 semesters of 10-500 Senior Research Project, totaling 24 units and counting as 2 electives.

10-500	Senior Research Project	24

Artificial Intelligence Two-Course Sequence

The Artificial Intelligence sequence requires 15-381 plus another course from the Artificial Intelligence sequence list.

Required introductory course:		
15-381	Artificial Intelligence: Representation and Problem Solving	9
Plus one of:		
15-388	Practical Data Science	9
17-537	Artificial Intelligence Methods for Social Good	9

Bioinformatics Two-Course Sequence

Students interested in the Bioinformatics sequence can choose between the Computational Genomics pair or the Biological Modeling pair.

Take both courses in Computational Genomics:		
02-510	Computational Genomics	Var.
03-511	Computational Molecular Biology and Genomics	9
Or take both courses in Biological Modeling:		
02-530	Cell and Systems Modeling	12
03-512	Computational Methods for Biological Modeling and Simulation	9

Computation, Organizations, and Society Two-Course Sequence

Students take both courses in the Computation, Organizations, and Society sequence. Please be aware that both of these courses are offered only intermittently.

17-621	Computational Modeling of Complex Socio-Technical Systems	12
17-685	Dynamic Network Analysis	12

Computer Vision Two-Course Sequence

Students have the choice between one of two introductory courses (16-311 or 16-385) plus another advanced course from the Computer Vision sequence.

Take one introductory course:		
16-311	Introduction to Robotics	12
or 16-385	Computer Vision	
Plus one of:		
15-463	Computational Photography	12
16-720	Computer Vision	12
16-725	Medical Image Analysis	12
16-823	Physics-based Methods in Vision (Appearance Modeling)	12
16-824	Visual Learning and Recognition	12

Language Technologies Two-Course Sequence

The Language Technologies sequence requires 11-411 plus another course from the Language Technologies sequence list.

Required introductory course:		
11-411	Natural Language Processing	12
Plus one of:		
11-441	Machine Learning for Text Mining	9
11-442	Search Engines	9
11-661	Language and Statistics	12
11-731	Machine Translation and Sequence-to-Sequence Models	12
11-751	Speech Recognition and Understanding	12
11-755	Machine Learning for Signal Processing	12
11-763	Structured Prediction for Language and other Discrete Data	12

Neural Cognition Two-Course Sequence

Students can take any two courses from the Neural Cognition sequence.

15-386	Neural Computation	9
15-883	Computational Models of Neural Systems	12

36-759	Statistical Models of the Brain	12
85-419	Introduction to Parallel Distributed Processing	9

Public Policy Two-Course Sequence

Students take both courses in the Public Policy sequence. If interested in this option, students should contact the Machine Learning Minor Director to confirm that 10-831 Special Topics in Machine Learning and Policy will be offered in an appropriate semester. Also, note that these two courses combine to count as only one elective, since they are under 9 units each.

Take both of:		
10-830	Machine Learning in Policy	12
10-831	Special Topics in Machine Learning and Policy	6

Robotics Two-Course Sequence

The Robotics sequence requires 16-311 plus another course from the Robotics sequence list.

Required introductory course:		
16-311	Introduction to Robotics	12
Plus one of:		
16-745	Dynamic Optimization	12
16-831	Statistical Techniques in Robotics	12
16-899	Special Topics Section C: Adaptive Control and Reinforcement Learning	12

Double Counting

No course in the Machine Learning Minor may be counted towards another SCS minor. Additionally, at least 3 courses (each being at least 9 units) must be used for only the Machine Learning Minor, not for any other major or minor. (These double counting restrictions apply specifically to the Core Courses and the Electives. Prerequisites may be counted towards other SCS minors and do not count towards the 3 courses that must be used for only the Machine Learning Minor.)

GRADES

The core courses (10-401/10-601 and 36-401) must average to at least a 3.0 (i.e., 1 A and 1 C, or 2 Bs). All courses for the Machine Learning Minor, including prerequisites, must be passed with at least a C. The student's overall, university-wide QPA must remain at least 2.5.

ADMISSION

The Machine Learning Minor is open to undergraduate students in any major at Carnegie Mellon. (SCS students should consult with their academic advisor for the existence of a machine learning concentration.) Students should apply for admission at least one semester before their expected graduation date, but are encouraged to apply as soon as they have taken the prerequisite classes for the minor. The application can be found on the Machine Learning Minor website.

Neural Computation Minor

Director: Dr. Tai Sing Lee
Administrative Coordinator: Melissa Stupka
Website: http://www.cnbc.cmu.edu/upnc/nc_minor/

Neural computation is a scientific enterprise to understand the neural basis of intelligent behaviors from a computational perspective. Study of neural computation includes, among others, decoding neural activities using statistical and machine learning techniques, and developing computational theories and neural models of perception, cognition, motor control, decision-making and learning. The neural computation minor allows students to learn about the brain from multiple perspectives, and to acquire the necessary background for graduate study in neural computation. Students enrolled in the minor will be exposed to, and hopefully participate in, the research effort in neural computation and computational neuroscience at Carnegie Mellon University.

The minor in Neural Computation is an intercollege minor jointly sponsored by the School of Computer Science, the Mellon College of Science, and the Dietrich College of Humanities and Social Sciences, and is coordinated by the Center for the Neural Basis of Cognition (CNBC) (http://www.cnbc.cmu.edu).

The Neural computation minor is open to students in any major of any college at Carnegie Mellon. It seeks to attract undergraduate students from computer science, psychology, engineering, biology, statistics, physics, and mathematics from SCS, CIT, H&SS and MCS.

The Neural Computation minor is open to students in any major of any college at Carnegie Mellon. It seeks to attract undergraduate students from computer science, psychology, engineering, biology, statistics, physics, and mathematics from SCS, CIT, Dietrich College and MCS. The primary objective of the minor is to encourage students in biology and psychology to take computer science, engineering and mathematics courses, to encourage students in computer science, engineering, statistics and physics to take courses in neuroscience and psychology, and to bring students from different disciplines together to form a community. The curriculum and course requirements are designed to maximize the participation of students from diverse academic disciplines. The program seeks to produce students with both basic computational skills and knowledge in cognitive science and neuroscience that are central to computational neuroscience.

APPLICATION

tudents must apply for admission no later than November 30 of their senior years; an admission decision will usually be made within one month. Students are encouraged to apply as early as possible in their undergraduate careers so that the director of the Neural Computation minor can provide advice on their curriculum, but should contact the program director any time even after the deadline.

To apply, send email to the director of the Neural Computation minor Dr. Tai Sing Lee (tai@cnbc.cmu.edu) (tai@cnbc.cmu.edu) and copy Melissa Stupka (mstupka@cnbc.cmu.edu) (mstupka@cnbc.cmu.edu). Include in your email:

- Full name
- Andrew ID
- Preferred email address (if different)
- Your class and College/School at Carnegie Mellon
- Semester you intend to graduate
- All (currently) declared majors and minors
- Statement of purpose (maximum 1 page) – Describes why you want to take this minor and how it fits into your career goals
- Proposed schedule of required courses for the Minor (this is your plan, NOT a commitment)
- Research projects you might be interested in

Curriculum

The Minor in Neural Computation will require a total of five courses: four courses drawn from the four core areas (A: neural computation, B: neuroscience, C: cognitive psychology, D: intelligent system analysis), one from each area, and one additional depth elective chosen from one of the core areas that is outside the student's major. The depth elective can be replaced by a one-year research project in computational neuroscience. No more than two courses can be double counted toward the student's major or other minors. However, courses taken for general education requirements of the student's degree are not considered to be double counted. A course taken to satisfy one core area cannot be used to satisfy the course requirement for another core area. The following listing presents a set of current possible courses in each area. Other computational neuroscience courses are being developed at Carnegie Mellon and University of Pittsburgh that will also satisfy core area A requirement and the requirements will be updated as they come on-line. Substitution is possible but requires approval.

A. Neural Computation

		Units
15-386	Neural Computation	9
15-387	Computational Perception	9
15-883	Computational Models of Neural Systems	12
85-419	Introduction to Parallel Distributed Processing	9
86-375	Computational Perception	9
Pitt-Mathematics-1800 Introduction to Mathematical Neuroscience		9

B. Neuroscience

03-362	Cellular Neuroscience	9
03-363	Systems Neuroscience	9
03-761	Neural Plasticity	9
42-630	Introduction to Neuroscience for Engineers (crosslisted with 18-690)	12
85-765	Cognitive Neuroscience	Var.
Pitt-Neuroscience 1000 Introduction to Neuroscience		9

C. Cognitive Psychology

85-211	Cognitive Psychology	9
85-213	Human Information Processing and Artifical Intelligence	9

85-412	Cognitive Modeling	9
85-419	Introduction to Parallel Distributed Processing	9
85-426	Learning in Humans and Machines	9
85-765	Cognitive Neuroscience	Var.

D. Intelligent System Analysis

10-601	Introduction to Machine Learning (Masters)	12
15-381	Artificial Intelligence: Representation and Problem Solving	9
15-386	Neural Computation	9
15-387	Computational Perception	9
15-494	Cognitive Robotics: The Future of Robot Toys	12
16-299	Introduction to Feedback Control Systems	12
16-311	Introduction to Robotics	12
16-385	Computer Vision	9
18-290	Signals and Systems	12
24-352	Dynamic Systems and Controls	12
36-225	Introduction to Probability Theory	9
36-247	Statistics for Lab Sciences	9
36-401	Modern Regression	9
36-410	Introduction to Probability Modeling	9
36-746	Statistical Methods for Neuroscience and Psychology	12
42-631	Neural Data Analysis	9
42-632	Neural Signal Processing	12
86-375	Computational Perception	9
86-631	Neural Data Analysis	9

Prerequisites

The required courses in the above four core areas require a number of basic prerequisites: basic programming skills at the level of 15-110 Principles of Computing and basic mathematical skills at the level of 21-122 Integration and Approximation or their equivalents. Some courses in Area D require additional prerequisites. Area B Biology courses require, at minimum, 03-121 Modern Biology. Students might skip the prerequisites if they have the permission of the instructor to take the required courses. Prerequisite courses are typically taken to satisfy the students' major or other requirements. In the event that these basic skill courses are not part of the prerequisite or required courses of a student's major, one of them can potentially count toward the five required courses (e.g. the depth elective), conditional on approval by the director of the minor program.

Research Requirements (Optional)

The minor itself does not require a research project. The student however may replace the depth elective with a year-long research project. In special circumstances, a research project can also be used to replace one of the five courses, as long as (1) the project is not required by the student's major or other minor, (2) the student has taken a course in each of the four core areas (not necessarily for the purpose of satisfying this minor's requirements), and (3) has taken at least three courses in this curriculum not counted toward the student's major or other minors. Students interested in participating in the research project should contact any faculty engaged in computational neuroscience or neural computation research at Carnegie Mellon or in the University of Pittsburgh. A useful webpage that provides listing of faculty in neural computation is www.cnbc.cmu.edu/computational-neuroscience. The director of the minor program will be happy to discuss with students about their research interest and direct them to the appropriate faculty.

Fellowship Opportunities

The Program in Neural Computation (PNC) administered by the Center for the Neural Basis of Cognition currently provides 3-4 competitive full-year fellowships ($11,000) to Carnegie Mellon undergraduate students to carry out mentored research in neural computation. The fellowship has course requirements similar to the requirements of the minor. Students do not apply to the fellowship program directly. They have to be nominated by the faculty members who are willing to mentor them. Therefore, students interested in the full-year fellowship program should contact and discuss research opportunities with any CNBC faculty at Carnegie Mellon or University of Pittsburgh working in the area of neural computation or computational neuroscience and ask for their nomination by sending email to Dr. Tai Sing Lee, who also administers the undergraduate fellowship program at Carnegie Mellon. See www.cnbc.cmu.edu/training/undergraduate/undergraduate-research-fellowships-in-computational-neuroscience/ for details.

The Program in Neural Computation also offers a summer training program for undergraduate students from any U.S. undergraduate college. The students will engage in a 10-week intense mentored research and attend a series of lectures in neural computation. See www.cnbc.cmu.edu/training/undergraduate/summer-undergraduate-research-program-in-computational-neuroscience/ for application information.

Robotics Additional Major

Director: Dr. Howie Choset
Administrative Coordinator: Barbara (B.J.) Fecich
Website: http://addlmajor.ri.cmu.edu/

The Additional Major in Robotics focuses on the theme that robotics is both multidisciplinary and interdisciplinary. This means that it draws from many fields, such as mechanical engineering, computer science and electrical engineering, and it also integrates these fields in a novel manner. The foundation of this program lies in motion and control. Upon this base, sensing, cognition, and action are layered. Since robotics involves building artifacts that embody these fundamentals, foci, and systems thinking, there is a "hands-on" course requirement. These foci are brought together by a unique systems perspective special to robotics. Students will complete a capstone course that will tie together previously learned skills and knowledge.

Admission

The Additional Major in Robotics is available to all Carnegie Mellon undergraduate students. Students should apply for the Robotics Additional Major their freshman year. Students in their sophomore year may apply, provided they meet the requirements and their schedule can accommodate the courses. The application is available via the program website and is due early February. Decisions on admittance to the Additional Major will be emailed to students in time for Fall registration. Application materials include:

- Full name and email address
- Home college, expected graduation date, and list of all declared Majors and Minors
- Statement of purpose (maximum 1 page, single spaced, to articulate why the student wants to pursue the Robotics Additional Major)
- Proposed schedule of required courses
- Unofficial Academic Record (can be downloaded from SIO)

Curriculum

Prerequisites

Calculus		Units
21-259	Calculus in Three Dimensions	9
Linear Algebra (choose one)		
18-202	Mathematical Foundations of Electrical Engineering	12
21-240	Matrix Algebra with Applications	10
21-241	Matrices and Linear Transformations	10
21-260	Differential Equations	9
24-311	Numerical Methods	12
Programming in C		
15-122	Principles of Imperative Computation	10
or knowledge and experience programming in C		

Required Courses

Choose 10 courses total (one from each category plus two electives):

Overview		Units
16-311	Introduction to Robotics	12
Controls		
06-464	Chemical Engineering Process Control	9
16-299	Introduction to Feedback Control Systems	12
18-370	Fundamentals of Control	12
24-451	Feedback Control Systems	12
16-xxx	Upper-level RI course with instructor and Program Director's permission	9-12
Kinematics		
16-384	Robot Kinematics and Dynamics	12

16-xxx	Upper-level RI course with instructor and Program Director's permission	9-12

Machine Perception

15-463	Computational Photography	12
16-385	Computer Vision	9
16-421	Vision Sensors	12
16-423	Designing Computer Vision Apps	12
85-370	Perception	9
85-395	Applications of Cognitive Science	9
16-xxx	Upper-level RI course with instructor and Program Director's permission	9-12

Cognition and Reasoning

10-401	Introduction to Machine Learning (Undergrad)	12
or 10-601	Introduction to Machine Learning (Masters)	
11-344	Machine Learning in Practice	12
15-381	Artificial Intelligence: Representation and Problem Solving	9
15-494	Cognitive Robotics: The Future of Robot Toys	12
16-xxx	Upper-level RI course with instructor and Program Director's permission	9-12

"Hands-on Course"

15-491	Special Topic: CMRoboBits: AI and Robots for Daily-Life Problems	12
16-362	Mobile Robot Algorithms Laboratory	12
16-423	Designing Computer Vision Apps	12
18-349	Introduction to Embedded Systems	12
18-578	Mechatronic Design	12
16-xxx	Upper-level RI project course e.g., 16-861 or 16-865 or independent study with instructor and Program Director's permission	9-12

Systems Engineering

16-450	Robotics Systems Engineering	12

Capstone Course

16-474	Robotics Capstone	12

Required Electives (choose two)

10-401	Introduction to Machine Learning (Undergrad)	12
or 10-601	Introduction to Machine Learning (Masters)	
11-344	Machine Learning in Practice	12
15-381	Artificial Intelligence: Representation and Problem Solving	9
15-424	Logical Foundations of Cyber-Physical Systems	12
15-462	Computer Graphics	12
15-463	Computational Photography	12
15-491	Special Topic: CMRoboBits: AI and Robots for Daily-Life Problems	12
15-494	Cognitive Robotics: The Future of Robot Toys	12
16-264	Humanoids	12
16-362	Mobile Robot Algorithms Laboratory	12
16-385	Computer Vision	9
16-421	Vision Sensors	12
16-423	Designing Computer Vision Apps	12
16-597	Undergraduate Reading and Research	Var.
18-342	Fundamentals of Embedded Systems	12
18-349	Introduction to Embedded Systems	12
18-578	Mechatronic Design	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

Students may count up to 12 units of 16-597 Undergraduate Reading and Research towards the major requirements. A student can also take additional courses from the core; e.g., a student who takes 16-385 as a core can take 16-421 as an elective.

Graduate level Robotics courses may be used to meet elective requirement with permission from the Program Director. 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 Program Director.

A 3.0 QPA in the Additional Major curriculum is required for graduation. Courses that are taken Pass/Fail or audited cannot be counted for the Additional Major.

Double-Counting Restriction

Students are permitted to double count a maximum of six courses from their Primary Major towards the Additional Major in Robotics. CS Majors are permitted to double count a maximum of five courses from their Primary Major towards the Additional Major in Robotics.

Robotics Minor

Director: Dr. Howie Choset
Administrative Coordinator: Barbara (B.J.) Fecich
Website: http://undergrad.ri.cmu.edu/academics/minor/

The Minor in Robotics provides an opportunity for undergraduate students at Carnegie Mellon to learn the principles and practices of robotics through theoretical studies and hands-on experience with robots. The Minor is open to students in any major of any college at Carnegie Mellon. Students initially learn the basics of robotics in an introductory robotics overview course. Additional required courses teach control systems and robotic manipulation. Students also choose from a wide selection of electives in robotics, perception, 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 research for those considering graduate studies.

All Robotics Minors are required to take Introduction to Robotics (16-311). This course is designed to help students understand the big picture of what is going on in robotics through topics such as kinematics, mechanisms, motion planning, sensor based planning, mobile robotics, sensors, and vision. The minor also requires students to take a controls class and a kinematics class. These courses provide students with the necessary intuition and technical background to move on to more advanced robotics courses. In addition to the required courses, students must take 2 electives. The student must have course selection approved by the Director during the application submission process.

A 2.5 QPA in the Minor curriculum is required for graduation. Courses that are taken Pass/Fail or audited cannot be counted for the Minor.

Admission

Admission to the Undergraduate Minor in Robotics is limited to current Carnegie Mellon students. Students interested in signing up for the minor should fill out the application form available on the program website.

Prerequisite

Successful candidates for the Robotics Minor will have prerequisite knowledge of C language, basic programming skills, and familiarity with basic algorithms. Students can gain this knowledge by taking 15-122 Principles of Imperative Computation.

Required Courses

Overview:		Units
16-311	Introduction to Robotics	12

Controls (choose one of the following):

06-464	Chemical Engineering Process Control	9
24-451	Feedback Control Systems	12
18-370	Fundamentals of Control	12
16-299	Introduction to Feedback Control Systems (Computer Science)	12
16-xxx	Upper-level RI course with instructor and Program Director's permission	

Kinematics (choose one of the following):

16-384	Robot Kinematics and Dynamics	12
16-xxx	Upper-level RI course with instructor and Program Director's permission	

Electives

Two Electives (chosen from the following):		Units
10-401	Introduction to Machine Learning (Undergrad) (or 10-601 Introduction to Machine Learning)	12

11-344	Machine Learning in Practice	12
15-381	Artificial Intelligence: Representation and Problem Solving	9
15-424	Logical Foundations of Cyber-Physical Systems	12
15-462	Computer Graphics	12
15-463	Computational Photography	12
15-491	Special Topic: CMRoboBits: AI and Robots for Daily-Life Problems	12
15-494	Cognitive Robotics: The Future of Robot Toys	12
16-264	Humanoids	12
16-362	Mobile Robot Algorithms Laboratory	12
16-385	Computer Vision	9
16-421	Vision Sensors	12
16-423	Designing Computer Vision Apps	12
16-597	Undergraduate Reading and Research	Var.
18-342	Fundamentals of Embedded Systems	12
18-349	Introduction to Embedded Systems	12
18-578	Mechatronic Design	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

Graduate level Robotics courses may be used to meet the elective requirement with permission from the Program Director. 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 Program Director.

Students may count up to 12 units of 16-597 Undergraduate Reading and Research towards the minor requirements.

Double-Counting Restriction

Courses being used to satisfy the requirements for the Robotics Minor may not be counted towards another minor. Students are permitted to double count a maximum of two courses from their Major (excluding General Education requirements) towards the Minor in Robotics. Free electives are not subject to the double counting policy.

Software Engineering Minor

Director: Claire Le Goues (clegoues@cs.cmu.edu)
Website: http://isri.cmu.edu/education/undergrad/

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.

The Software Engineering minor 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.

Admission

The Software Engineering Minor is open to undergraduate students in any major in the university. (SCS students should consult with their academic advisor for the existence of a Software Engineering concentration.) We encourage students to submit applications no later than 3 days before the beginning of the Spring and Fall course registration periods, so that subsequent decisions can help students plan their course schedules effectively. However, students may petition the Director for admission outside this general schedule.

To apply, send the directors an email. Include in your email:

- Full name
- Andrew ID
- Preferred email address (if different)
- Semester you intend to graduate
- QPA
- All (currently) declared majors and minors, or home college if no major declared
- Statement of purpose (maximum 1 page) - Describes why you want to take this minor and how it fits into your career goals
- Proposed schedule of required courses and internship (this is your plan, NOT a commitment)

Prerequisite

		Units
17-214	Principles of Software Construction: Objects, Design, and Concurrency	12

Core Course Requirements

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

Electives

The minor requires three elective courses, one selected from each of the following categories:

1. One domain-independent course focused on technical software engineering material:

15-414	Bug Catching: Automated Program Verification	9
17-355	Program Analysis	12
17-356	Software Engineering for Startups	12
17-615	Software Process Definition	9
17-651	Models of Software Systems	12
17-652	Methods: Deciding What to Design	12
17-653	Managing Software Development (prereq: 17-413 or an internship)	12
17-654	Analysis of Software Artifacts	12
17-655	Architectures for Software Systems (prereq: 17-413 or an internship)	12
Other Software Engineering graduate classes may be taken; you must get preapproval from the program director prior to taking the class.		

2. One engineering-focused course with a significant software component:

15-410	Operating System Design and Implementation	15
15-412	Operating System Practicum	Var.
15-440	Distributed Systems	12
15-441	Computer Networks	12
17-437	Web Application Development	12
17-643	Hardware for Software Engineers	Var.
18-649	Distributed Embedded Systems	12
Other courses may be acceptable; you must get preapproval from the program director prior to taking the course.		

3. One course that explores computer science problems related to existing and emerging technologies and their associated social, political, legal, business, and organizational contexts:

15-390	Entrepreneurship for Computer Science	9
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
19-402	Telecommunications Technology and Policy for the Internet Age	12
19-403	Policies of Wireless Systems	12
70-311	Organizational Behavior	9
70-414	Entrepreneurship for Engineers	9
70-421	Entrepreneurship for Computer Scientists	9
70-471	Supply Chain Management	9
88-341	Team Dynamics and Leadership	9

Required Internship and Reflection Course

A software engineering internship of a minimum of 8 full-time weeks in an industrial setting is required. The student must be integrated into a team

and exposed to industry pressures. The intern may work in development, management, quality assurance, or other relevant positions. The director of the SE minor program has sole discretion in approving an internship experience based on these criteria. Students should confirm that an internship position is appropriate before accepting it, but internships that fulfill the criteria will also be accepted after the fact.

17-415	Software Engineering Reflection	6

Double Counting Rule

At most 2 of the courses used to fulfill the minor requirements may be counted towards any other major or minor program. This rule does not apply to 17-214 (a prerequisite for the minor) or courses counted for general education requirements.

The School of Computer Science will begin to offer 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.

Concentrations will be introduced during the 2018-19 academic year. Consult the SCS undergraduate programs website (https://www.cs.cmu.edu/undergraduate-programs) 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.

Security & Privacy Concentration

Lujo Bauer, Concentration Coordinator (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.

How to Apply

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. Students who complete the concentration can contact the concentration coordinator to receive a certificate attesting to their successful completion.

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 Students who have successfully completed 15-487 or 18-487 in Fall 2017 will be allowed to count that course as having satisfied this requirement for the concentration as long as they also successfully complete 17-333 (previously 08-533).	12

Theoretical Foundations course (choose one option):		Units
15-503	Introduction to Cryptography	9
or the following two courses:		
18-733	Applied Cryptography	12
18-734	Foundation of Privacy	12

System Design course (choose one):		Units
15-316	Software Foundations of Security and Privacy	9
18-732	Secure Software Systems	12

Usability or Policy course (select one):		Units
17-334	Usable Privacy and Security	9
or one of:		
17-333	Privacy Policy, Law, and Technology	9
18-734	Foundation of Privacy (if not used for the Theoretical Foundations requirement)	12

Depth course (complete one option below):	Units
Complete an elective course or at least 9 units of independent study in the security or privacy area. Consult with the concentration coordinator for elective options.	9
Complete five, rather than four, courses from the list above to satisfy the requirements described above (this might be achieved by taking both a policy and a usability course, or taking the two-course foundations alternative).	9-12

Prior Coursework

Any courses from the core or elective list successfully completed before Fall 2018 will likely also count toward concentration requirements, but check with the concentration coordinator to make sure your previous courses will count.

Anti-requisites

When two (or more) courses overlap significantly in the material they cover, only one can count toward the security and privacy concentration. Below is a list of anti-requisites; each bullet is a list of courses out of which only one can count toward the security and privacy concentration.

- Software Foundations of Security and Privacy (15-316)
  Secure Software Systems (18-732)
- Introduction to Cryptography (15-503)
  Applied Cryptography (18-733)

Excluded Courses

The following 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.

- Information Security and Privacy (17-331, previously 15-421, or equivalent crosslisted courses)
- Introduction to Computer Security (18-730)