School of Computer Science

Martial Hebert, Dean
Thomas Cortina, Associate Dean for Undergraduate Programs
Veronica Peet, Assistant Dean for Undergraduate Experience

Location: GHC 4115
www.cs.cmu.edu/undergraduate-programs (http://www.cs.cmu.edu/undergraduate-programs/)

Carnegie Mellon founded one of the first Computer Science departments in the world in 1965. As research and teaching in computing grew at a tremendous pace at Carnegie Mellon, the university formed the School of Computer Science (SCS) at the end of 1988. Carnegie Mellon was one of the first universities to elevate Computer Science into its own academic college at the same level as the Mellon College of Science and the College of Engineering. Today, SCS consists of seven departments and institutes, including the Computer Science Department that started it all, along with the Human-Computer Interaction Institute, the Institute for Software Research, the Computational Biology Department, the Language Technologies Institute, the Machine Learning Department, and the Robotics Institute. Together, these units make SCS a world leader in research and education. A few years ago, SCS launched two new undergraduate majors in Computational Biology and Artificial Intelligence (the first of its kind in the United States), and this year, SCS begins a fourth undergraduate major in Human-Computer Interaction. These new majors, along with the highly-ranked Computer Science major, give students in SCS distinct paths in the field of computing with ample opportunities in industry and advanced research.

The School of Computer Science offers the following majors and minors:
- B.S. in Artificial Intelligence
- B.S. in Computational Biology
- B.S. in Computer Science
- B.S. in Human-Computer Interaction
- Bachelor's in Computer Science and Art (joint with the College of Fine Arts)
- Additional major in Computational Biology
- Additional major in Computer Science
- Additional major in Human-Computer Interaction
- Additional major in Robotics
- Minor in Computer Science
- Minor in Computational Biology
- Minor in Human-Computer Interaction
- Minor in Language Technologies
- Minor in Machine Learning
- Minor in Neural Computation
- Minor in Robotics
- Minor in Software Engineering

Information for these majors and minors can be found through the navigation menu or through the links below:
- Artificial Intelligence (http://coursecatalog.web.cmu.edu/schools-colleges/schoolofcomputerscience/artificialintelligence/) (B.S. degree)
- Computational Biology (http://coursecatalog.web.cmu.edu/schools-colleges/schoolofcomputerscience/undergraduatecomputationalbiology/) (B.S. degree, additional major, minor)
- Computer Science (http://coursecatalog.web.cmu.edu/schools-colleges/schoolofcomputerscience/undergraduatecomputer.science/) (B.S. degree, additional major, minor)
- Human-Computer Interaction (http://coursecatalog.web.cmu.edu/schools-colleges/schoolofcomputerscience/humancomputerinteractionprogram/) (B.S. degree, additional interdisciplinary major, minor)
- SCS additional majors and minors (http://coursecatalog.web.cmu.edu/schools-colleges/schoolofcomputerscience/additionalmajorsandminors/)

Students who apply to, and are directly admitted into, the School of Computer Science can choose between four primary majors: Artificial Intelligence, Computational Biology, Computer Science and Human-Computer Interaction. Students with artistic and computing interests may be given the option to pursue a major in Computer Science and Art. Suitably prepared students from other Carnegie Mellon colleges are eligible to apply for internal transfer to the School of Computer Science and will be considered for transfer if grades in specific requirements are sufficiently high and space is available. Consult the program websites for specific requirements for transfer requests. Computation-oriented programs are also available within the Mellon College of Science, the Dietrich College of Humanities and Social Sciences, the College of Engineering and the College of Fine Arts.

Policies & Procedures

Academic Standards and Actions

Grading Practices
Grades given to record academic performance in SCS are detailed under Grading Practices at Undergraduate Academic Regulations (http://coursecatalog.web.cmu.edu/servicesandoptions/undergraduateacademicregulations/).

Dean’s List WITH HIGH HONORS
SCS recognizes each semester those undergraduates who have earned outstanding academic records by naming them to the Dean’s List with High Honors. The criterion for such recognition is a semester quality point average of at least 3.75 while completing a minimum of 36 factorable units and earning no incomplete grades.

Academic Actions
In the first year, quality point averages below 1.75 in either semester invoke an academic action. For all subsequent semesters an academic action will be taken if the semester quality point average or the cumulative quality point average (excluding the first year) is below 2.00.

Probation: The action of probation will be taken in the following cases based on QPA:
1. One semester of the first year is below 1.75 QPA;
2. The semester QPA of a student in good standing beyond the first year falls below 2.00.

The term of probation is one semester as a full-time student. First year students are no longer on probation at the end of the second semester if the second semester’s QPA and the cumulative QPA is 1.75 or above. Students in the third or subsequent semester of study are no longer on probation at the end of one semester if the semester QPA and cumulative QPA (excluding the first year) are 2.00 or above.

Probation Continued: A student who has had one semester on probation and is not yet meeting minimum requirements but whose record indicates that the standards are likely to be met at the end of the next semester of study may be continued on probation, based on advisor recommendation. This action is normally taken only when a student’s semester QPA is above 2.0 but their cumulative QPA is not yet above 2.0.

Suspension: A student who does not meet minimum standards based on QPA at the end of one semester of probation can be suspended:
1. A first year student will be suspended if the QPA from each semester is below 1.75;
2. A student on probation in the third or subsequent semester of study will be suspended if the semester QPA is below 2.00.

The minimum period of suspension is one academic year (two non-summer semesters). Suspension is meant to allow a student to take a pause from their academic studies to address the issues that are causing poor academic performance. At the end of that period a student may return to campus (on probation) by:
1. completing a Return from Leave form from the HUB, and
2. submitting an additional written statement to the SCS Associate Dean for Undergraduate Programs, minimum one page, that outlines what the student did while on leave to address the issues that led to the suspension and that would indicate future success on return, and
3. (optional) submitting up to two letters of support from individuals supporting the student’s return, and
4. written approval from the student’s academic advisor and the Associate Dean for Undergraduate Programs in consultation with the Office of Student Affairs and the Office of International Education as appropriate.

Students who have been suspended or have withdrawn are required to absent themselves from the campus (including residence halls and Greek houses) within a maximum of two days after the action and to remain off the campus for the duration of the time specified. This action includes
debarment from part-time or summer courses at the university for the duration of the period of the action. Although suspended students may not hold student jobs, students on academic suspension may, under certain circumstances, have a non-student job with the university. Students on disciplinary or administrative suspension may not.

**Drop:** This is a permanent severance. Students who have been suspended and who fail to meet minimum standards in the semester that they return to school on probation will be dropped from the School of Computer Science. Students who have been dropped and are not admitted to another program at the university are required to absent themselves from the campus (including residence halls and Greek houses) within a maximum of two days after the action.

**Appeal:** Students may appeal a suspension or drop decision in writing within 10 days of notification if, under consultation with their academic advisor, they have information that would indicate that they can continue in the next semester and exit their probationary status. Instructions on the appeal process are given in the suspension letter that is sent to the student.

**Other Actions:** In addition to academic actions based on QPA, the Associate Dean for Undergraduate Programs may place students on probation, subsequent suspension or drop, if they do not demonstrate reasonable progress through the core curriculum of their major (e.g., not completing a core class after 3 attempts, not completing the required 100-level core courses by the end of the sophomore year, etc.). Students are encouraged to consult with their academic advisor about any concerns with regard to lack of progress in their chosen SCS major to determine if any course drop or withdrawal will lead to an action.

The relation indicated above between probation, suspension and drop is nominal. In unusual circumstances, SCS College Council may suspend or drop a student without prior probation.

---

**Leave of Absence and Return from Leave of Absence**

SCS undergraduate students may elect to take a leave of absence for a variety of reasons, after consultation with their academic advisor. Students who wish to take a leave of absence must do so by the last day of classes before final exams begin and before final grades are posted (in case this is earlier). Students requesting a leave of absence must complete a form from the HUB and have this signed by their academic advisor and SCS Associate Dean for Undergraduate Programs. Students who take a leave of absence up to the last day to drop classes will have all of their classes dropped. Students who take a leave of absence after the last day to drop classes will be assigned a grade of W (withdrawal) for all of their classes.

Students returning from a leave of absence are required to submit a Return from Leave of Absence form to their academic advisor for approval by the student's academic advisor and the SCS Associate Dean for Undergraduate Programs. In addition, the student must also supply a letter that explains the reason for the leave, the actions that were performed during the leave to prepare the student for a successful return, and a description of the student's campus resources, if required, that would be used by the student in order to increase the likelihood of success. Students returning from a leave are also encouraged to provide up to two letters of support from people close to the student (e.g., family, friends, clergy, teachers, coaches, others as appropriate). Requests to return are reviewed by the student's academic advisor, the Associate Dean and the Student Affairs liaison to determine eligibility and any resources that need to be put into place to assist the student upon return. Contact the SCS Undergraduate Office (GHC 4115) for more information.

---

**Internal Transfer within SCS**

First year students admitted to SCS are considered undeclared during their first year. These students declare their SCS major in the second semester of their first year of study. SCS students who wish to transfer from one SCS major to another SCS major may do so by applying for transfer by mid-semester break during the semester the transfer is desired. These students should consult with their academic advisor and the program director of the intended major for more information about specific course requirements and academic plans. Internal SCS transfers do not have any grade requirements. Transfers are approved based on demonstrated interest, ability, and available space in the intended major.

**Transfer into SCS / Dual-Degree**

Undergraduate students admitted to colleges at CMU other than SCS and wishing to transfer to Computer Science or pursue a dual degree in Computer Science should consult with the Director of the Computer Science major during their first year. Students wishing to transfer to Computational Biology or pursue a dual degree in Computational Biology should consult with the Assistant Department Head for Education in the Computational Biology Department during their first year. See the individual program pages for Computer Science (http:// coursescatalog.web.cmu.edu/schools-colleges/schooIOfcomputerscience/ undergraduatecomputerscience/) and Computational Biology (http:// coursescatalog.web.cmu.edu/schools-colleges/schooIOfcomputerscience/ undergraduatecomputationalbiology/) for locations.

- For the Computer Science major, students must complete 21-127 (or equivalent), 15-122, 15-150, 15-210, 15-213, 15-251 with an expected overall QPA over these six courses of 3.6 or higher and an overall QPA of at least 3.0 in order to be considered for transfer or dual degree.
- For the Computational Biology major, students must complete 21-127 (or equivalent), 15-122, 15-251, 15-351 (or 15-210*), 03-121 and 02-250 with an expected overall QPA over these six courses of 3.6 or higher and an overall QPA of at least 3.0 in order to be considered for transfer or dual degree. (*Students who take 15-210 will need to also take 15-150; this course is not required for the B.S. in Computational Biology but can count as an elective.)
- Consult the websites for the Human-Computer Interaction and Artificial Intelligence majors for guidelines for transfer.

Students may apply for transfer by the mid-semester break in the semester when the final course(s) of the six required courses will be completed. In the case of course(s) in progress, the mid-semester grades will be used in the QPA calculation. The decision to allow transfer or dual degree will be made by committee based on the student's academic performance (in the specified courses and in their courses overall if necessary), additional involvement in SCS and other computing-related activities, and availability of space in the student's class level. Students should consult the SCS Undergraduate Office for complete information concerning minimum requirements, instructions and deadlines.

---

**External Transfer**

A student currently enrolled at another university or college who wishes to transfer to SCS should first apply through the Office of Admission. If the Office of Admission believes the applicant meets admission guidelines, the student's record is sent to SCS for evaluation. Admission is based on seat availability, overall academic performance and course rigor from the student's current institution, ability to complete the rigorous SCS program on time, and the application material including recommendations and reflection essay(s). It is important to note that extremely few external transfers are admitted to the SCS program at Carnegie Mellon University due to space limitations.

**Graduation Requirements**

1. A requirement for graduation is the completion of the program specified for a degree with a cumulative quality point average of 2.00 or higher for all courses taken after the first year.
2. Students must be recommended for a degree by the faculty of SCS.
3. A candidate for the bachelor's degree must complete at the University a minimum of four semesters of full-time study, or the equivalent of part-time study, comprising at least 180 units of course work.
4. Students will be required to have met all financial obligations to the university before being awarded a degree.

A student who does not meet the QPA requirement above must petition SCS College Council for a waiver of the first requirement.

**General Education Requirements**

All undergraduate degrees(7,0),(990,990) in the School of Computer Science include depth in their particular field of study but also breadth through the general education requirements. General education requirements are part of SCS degrees to give students an opportunity to learn more about the world from scientific and humanistic points of view. These additional skills are useful for graduates since computing is often embedded in domains that are not entirely within the bounds of computing. SCS students will need to use their computing skills to solve problems alongside scientists and engineers, artists, social and cognitive scientists, historians, linguists, economists and business experts, and SCS students will need to communicate effectively and understand the ethical implications of their work. The general education requirements help SCS students gain this broad perspective so they can work well in a wide variety of domains.

---

**General Education Requirements**

All undergraduate degrees in the School of Computer Science include depth in their particular field of study but also breadth through the general education requirements. General education requirements are part of SCS degrees to give students an opportunity to learn more about the world from scientific and humanistic points of view. These additional skills are useful for graduates since computing is often embedded in domains that are not entirely within the bounds of computing. SCS students will need to use their computing skills to solve problems alongside scientists and engineers, artists, social and cognitive scientists, historians, linguists, economists and business experts, and SCS students will need to communicate effectively and understand the ethical implications of their work. The general education requirements help SCS students gain this broad perspective so they can work well in a wide variety of domains.

---

**General Education Requirements**

All undergraduate degrees in the School of Computer Science include depth in their particular field of study but also breadth through the general education requirements. General education requirements are part of SCS degrees to give students an opportunity to learn more about the world from scientific and humanistic points of view. These additional skills are useful for graduates since computing is often embedded in domains that are not entirely within the bounds of computing. SCS students will need to use their computing skills to solve problems alongside scientists and engineers, artists, social and cognitive scientists, historians, linguists, economists and business experts, and SCS students will need to communicate effectively and understand the ethical implications of their work. The general education requirements help SCS students gain this broad perspective so they can work well in a wide variety of domains.

---

**General Education Requirements**

All undergraduate degrees in the School of Computer Science include depth in their particular field of study but also breadth through the general education requirements. General education requirements are part of SCS degrees to give students an opportunity to learn more about the world from scientific and humanistic points of view. These additional skills are useful for graduates since computing is often embedded in domains that are not entirely within the bounds of computing. SCS students will need to use their computing skills to solve problems alongside scientists and engineers, artists, social and cognitive scientists, historians, linguists, economists and business experts, and SCS students will need to communicate effectively and understand the ethical implications of their work. The general education requirements help SCS students gain this broad perspective so they can work well in a wide variety of domains.

---

**General Education Requirements**

All undergraduate degrees in the School of Computer Science include depth in their particular field of study but also breadth through the general education requirements. General education requirements are part of SCS degrees to give students an opportunity to learn more about the world from scientific and humanistic points of view. These additional skills are useful for graduates since computing is often embedded in domains that are not entirely within the bounds of computing. SCS students will need to use their computing skills to solve problems alongside scientists and engineers, artists, social and cognitive scientists, historians, linguists, economists and business experts, and SCS students will need to communicate effectively and understand the ethical implications of their work. The general education requirements help SCS students gain this broad perspective so they can work well in a wide variety of domains.
Science and Engineering

All candidates for a B.S. degree in the School of Computer Science must complete a minimum of 36 units offered by the Mellon College of Science and/or the College of Engineering (CIT).

Computational Biology majors

For Computational Biology majors, consult the Computational Biology [program page](http://coursecatalog.web.cmu.edu/schools-colleges/computationalbiology/) for specific science and engineering requirements. The required science and engineering courses for the Computational Biology major also satisfy the General Education requirement for SCS by default.

Artificial Intelligence, Computer Science and Human-Computer Interaction majors

For Artificial Intelligence, Computer Science and Human-Computer Interaction majors, four courses in science and engineering are required, 9 units or more for each course, at least one course must have a laboratory component and at least two courses must be from the same department.

Non-lab courses that are usually taken by AI, CS and HCI majors to satisfy this requirement are given in the list below. (Consult your academic advisor for additional choices available each semester.)

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Course Title</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>02-223</td>
<td>Personalized Medicine: Understanding Your Own Genome</td>
<td>9</td>
</tr>
<tr>
<td>03-121</td>
<td>Modern Biology</td>
<td>9</td>
</tr>
<tr>
<td>03-125</td>
<td>Evolution</td>
<td>9</td>
</tr>
<tr>
<td>03-132</td>
<td>Basic Science to Modern Medicine</td>
<td>9</td>
</tr>
<tr>
<td>03-133</td>
<td>Neurobiology of Disease</td>
<td>9</td>
</tr>
<tr>
<td>06-100</td>
<td>Introduction to Chemical Engineering</td>
<td>12</td>
</tr>
<tr>
<td>06-221</td>
<td>Thermodynamics</td>
<td>9</td>
</tr>
<tr>
<td>09-105</td>
<td>Introduction to Modern Chemistry I</td>
<td>10</td>
</tr>
<tr>
<td>09-106</td>
<td>Modern Chemistry II</td>
<td>10</td>
</tr>
<tr>
<td>09-217</td>
<td>Organic Chemistry I</td>
<td>9</td>
</tr>
<tr>
<td>09-216</td>
<td>Organic Chemistry II</td>
<td>9</td>
</tr>
<tr>
<td>09-225</td>
<td>Climate Change: Chemistry, Physics and Planetary Science</td>
<td>9</td>
</tr>
<tr>
<td>12-100</td>
<td>Exploring CEE: Infrastructure and Environment in a Changing World</td>
<td>12</td>
</tr>
<tr>
<td>12-201</td>
<td>Geology</td>
<td>9</td>
</tr>
<tr>
<td>18-100</td>
<td>Introduction to Electrical and Computer Engineering</td>
<td>12</td>
</tr>
<tr>
<td>18-220</td>
<td>Electronic Devices and Analog Circuits</td>
<td>12</td>
</tr>
<tr>
<td>18-240</td>
<td>Structure and Design of Digital Systems</td>
<td>12</td>
</tr>
<tr>
<td>24-101</td>
<td>Fundamentals of Mechanical Engineering</td>
<td>12</td>
</tr>
<tr>
<td>24-231</td>
<td>Fluid Mechanics</td>
<td>10</td>
</tr>
<tr>
<td>24-261</td>
<td>Statics</td>
<td>10</td>
</tr>
<tr>
<td>24-351</td>
<td>Dynamics</td>
<td>10</td>
</tr>
<tr>
<td>24-358</td>
<td>Culinary Mechanics</td>
<td>9</td>
</tr>
<tr>
<td>27-215</td>
<td>Thermodynamics of Materials</td>
<td>12</td>
</tr>
<tr>
<td>27-324</td>
<td>Introduction to Polymer Science and Engineering</td>
<td>9</td>
</tr>
<tr>
<td>33-114</td>
<td>Physics of Musical Sound</td>
<td>9</td>
</tr>
<tr>
<td>33-120</td>
<td>Science and Science Fiction</td>
<td>9</td>
</tr>
<tr>
<td>33-121</td>
<td>Physics I for Science Students</td>
<td>12</td>
</tr>
<tr>
<td>or 33-141</td>
<td>Physics I for Engineering Students</td>
<td></td>
</tr>
<tr>
<td>or 33-151</td>
<td>Physics I for Engineering Students</td>
<td></td>
</tr>
<tr>
<td>33-142</td>
<td>Matter and Interactions</td>
<td>12</td>
</tr>
<tr>
<td>or 33-152</td>
<td>Matter and Interactions</td>
<td></td>
</tr>
<tr>
<td>33-224</td>
<td>Stars, Galaxies and the Universe</td>
<td>9</td>
</tr>
<tr>
<td>42-101</td>
<td>Introduction to Biomedical Engineering</td>
<td>12</td>
</tr>
<tr>
<td>42-202</td>
<td>Physiology</td>
<td>9</td>
</tr>
<tr>
<td>85-219</td>
<td>Biological Foundations of Behavior</td>
<td>9</td>
</tr>
</tbody>
</table>

At present, courses meeting the lab requirement include:

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Course Title</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>02-261</td>
<td>Quantitative Cell and Molecular Biology Laboratory</td>
<td>9</td>
</tr>
<tr>
<td>03-124</td>
<td>Modern Biology Laboratory</td>
<td>9</td>
</tr>
</tbody>
</table>
19-608 Privacy Policy, Law, and Technology 12
24-281 Introduction to Scientific Computing 2
24-311 Numerical Methods 12
27-410 Computational Techniques in Engineering 12
33-100 Basic Experimental Physics 6
33-115 Physics for Future Presidents 9
33-124 Introduction to Astronomy 9
33-232 Mathematical Methods of Physics 10
42-201 Professional Issues in Biomedical Engineering 3
49-300 Integrated Product Conceptualization 12

All Electrical and Computer Engineering graduate courses (18-6xx, 18-7xx, 18-8xx, 18-9xx) cannot be used for this requirement. Students interested in Engineering & Public Policy (19-xxx) courses that are not excluded above, particularly special topics courses, must consult with the CS Program Director to determine suitability for this requirement. In general, any MCS or CIT courses that are cross-listed with SCS courses or have significant mathematical or computational content cannot be used for this requirement. Students must consult with an SCS undergraduate advisor about any course to be used for the Science and Engineering requirement before registration.

Humanities and Arts

All candidates for a B.S. degree in the School of Computer Science must complete a minimum of 63 units offered by the College of Humanities & Social Sciences and/or the College of Fine Arts as prescribed below. Students pursuing a Bachelor's in Computer Science and Art (http://coursecatalog.web.cmu.edu/servicesandoptions/intercollegenew/schools/intercollege/#bcsacurriculumtext) should consult the general education requirements for that program.

A. Freshman Writing Requirement (9 units)

Complete one of the following writing options for 9 units:
76-101 Interpretation and Argument 9
76-102 Advanced First Year Writing: Special Topics (by invitation only)
or two of these three writing minis for 9 units total:
76-106 Writing about Literature, Art and Culture 4.5
76-107 Writing about Data 4.5
76-108 Writing about Public Problems 4.5

B. Breadth Requirement (minimum 27 units: 9 units each)

Complete three courses, one each from Category 1, Category 2, and Category 3. Students may use two minis totaling 9 units or more to satisfy one of the categories, with permission of the Associate Dean for Undergraduate Education, if the minis meet the goals of the desired category. NOTE: Artificial Intelligence majors replace Category 1 with Category 1A: Cognitive Studies which is a subset of Category 1.

Category 1 (for all SCS majors except Artificial Intelligence): Cognition, Choice and Behavior - this requirement explores the process of thinking, decision making, and behavior in the context of the individual.
70-311 Organizational Behavior 9
80-101 Dangerous Ideas in Science and Society 9
80-130 Introduction to Ethics 9
80-150 Nature of Reason 9
80-180 Nature of Language 9
80-221 Philosophy of Social Science 9
80-241 Ethical Judgments in Professional Life 9
80-242 Conflict and Dispute Resolution 9
80-270 Problems of Mind and Body: Meaning and Doing 9
80-271 Mind and Body: The Objective and the Subjective 9
80-275 Metaphysics 9
80-281 Language and Thought 9
80-330 Ethical Theory 9
85-102 Introduction to Psychology 9
85-211 Cognitive Psychology 9
85-213 Human Information Processing and Artificial Intelligence 9
85-221 Principles of Child Development 9
85-241 Social Psychology 9
85-251 Personality 9
85-261 Psychopathology 9
85-370 Perception 9
85-408 Visual Cognition 9
85-421 Language and Thought 9
88-120 Reason, Passion and Cognition 9
88-230 Human Intelligence and Human Stupidity 9

Category 1A (for Artificial Intelligence majors): Cognitive Studies - this requirement explores how the brain and the mind work.
85-211 Cognitive Psychology 9
85-213 Human Information Processing and Artificial Intelligence 9
85-370 Perception 9
85-408 Visual Cognition 9
85-421 Language and Thought 9

Category 2 (all SCS majors): Economic, Political and Social Institutions - this requirement explores the processes by which institutions organize individual preferences and actions into collective outcomes.
19-101 Introduction to Engineering and Public Policy 12
36-303 Sampling, Survey and Society 9
70-332 Business, Society and Ethics 9
70-364 Business Law 9
73-102 Principles of Microeconomics 9
73-103 Principles of Macroeconomics 9
79-189 History of Democracy: Thinking Beyond the Self 9
79-244 Women in American History 9
79-245 Capitalism and Individualism in American Culture 9
79-299 From Newton to the Nuclear Bomb: History of Science, 1750-1950 9
79-300 History of American Public Policy 9
79-310 U. S. Business History: 1870 to the Present 9
79-320 Women, Politics, and Protest 9
79-321 Documenting Human Rights 9
79-331 Body Politics: Women and Health in America 9
79-341 The Cold War in Documents and Film 9
79-383 The History of Capitalism 9
80-135 Introduction to Political Philosophy 9
80-136 Social Structure, Public Policy & Ethics 9
80-243 Ethics of Leadership 9
80-244 Environmental Ethics 9
80-245 Medical Ethics 9
80-324 Philosophy of Economics 9
80-335 Social and Political Philosophy 9
80-341 Computers, Society and Ethics 9
84-104 Decision Processes in American Political Institutions 9
84-275 Comparative Politics 9
84-310 International Political Economy 9
84-322 Nonviolent Conflict and Revolution 9
84-324 The Future of Democracy 9
84-326 Theories of International Relations 9
84-352 Representation and Redistricting 9
84-362 Diplomacy and Statecraft 9
84-380 US Grand Strategy 9
84-386 The Privatization of Force 9
84-389 Terrorism and Insurgency 9
84-390 Social Media, Technology, and Conflict 9
84-402 Judicial Politics and Behavior 9
84-405 The Future of Warfare 9
84-414 International and Subnational Security 9
88-284 Topics of Law: The Bill of Rights 9

Category 3 (all SCS majors): Cultural Analysis - this requirement seeks to recognize cultures that have shaped and continue to shape the human experience; courses in this category are usually either broad in place, time, or cultural diversity.
48-240 Historical Survey of World Architecture and Urbanism I 9
57-173 Survey of Western Music History 9
60-105 Critical Theory in Art I 9
humane/arts courses with lower units together to form a single course of 9 units or more. Students are encouraged, but not required, to take courses from different departments to gain additional breadth and to create new opportunities for engagement with the university community.

Deletions

Some courses from the Dietrich College or the College of Fine Arts may not count toward the unconstrained electives in Humanities and Arts in SCS due to the technical (computing and/or mathematical) nature of the courses. Courses from the following departments do not count toward the unconstrained Humanities and Arts electives:

- Statistics and Data Science (36), except 36-303 Sampling, Survey and Society
- Information Systems (67)
- Economics (73), except 73-102 Principles of Microeconomics and 73-103 Principles of Macroeconomics

Additionally, the following courses do not count toward the unconstrained Humanities and Arts electives:

51-257 Introduction to Computing for Creative Practices 10
51-327 Design Center: Introduction to Web Design 9
51-328 Design Center: Design for Digital Systems 9
76-388 Coding for Humans 9
76-481 Introduction to Multimedia Design 12
76-487 Web Design 12
80-110 Nature of Mathematical Reasoning 9
80-210 Logic and Proofs 9
80-211 Logic and Mathematical Inquiry 9
80-212 Arguments and Logical Analysis 9
80-222 Measurement and Methodology 9
80-223 Causality and Probability 9
80-305 Decision Theory 9
80-310 Formal Logic 9
80-311 Undecidability and Incompleteness 9
80-314 Causal Discovery, Statistics, and Machine Learning 9
80-315 Modal Logic 9
80-316 Logic and AI 9
80-317 Introduction to Ramsey Theory 6
80-405 Game Theory 9
80-411 Proof Theory 9
80-413 Category Theory 9
80-419 Interactive Theorem Proving 9
80-521 Seminar on Formal Epistemology: Belief and Evidence Var.
85-219 Biological Foundations of Behavior 9
85-310 Research Methods in Cognitive Psychology 9
85-314 Cognitive Neuroscience Research Methods 9
85-414 Cognitive Neuropsychology 9
85-426 Learning in Humans and Machines 9
88-251 Empirical Research Methods 9
88-372 Social and Emotional Brain 9

Additions

The following courses outside of Dietrich College and the College of Fine Arts may count toward the unconstrained Humanities and Arts electives:

16-161 ROB Freshman Seminar: Artificial Intelligence and Humanity 9
16-397 Art, Conflict and Technology 12
17-333 Privacy Policy, Law, and Technology 9
17-562 Law of Computer Technology 9
19-101 Introduction to Engineering and Public Policy 12
19-351 Applied Methods for Technology-Policy Analysis 9
19-402 Telecommunications Technology and Policy for the Internet Age 12
19-403 Policies of Wireless Systems 12
19-411 Science and Innovation Leadership for the 21st Century: Firms, Nations, and Tech 9
32-201 Leadership & Management 9
32-402 Leadership and Ethics 9
Honors Research Thesis

Students considering going on to graduate school in Computer Science or related disciplines should take a wide variety of Computer Science and Mathematics courses, as well as consider getting involved in independent research as early as possible. This would be no later than the junior year and can begin even earlier. Students interested in graduate school in computer science or its related areas are strongly encouraged to participate in the SCS Honors Undergraduate Research Thesis program. Additionally, graduate CS courses can be taken with permission of the instructor and in consultation with an academic advisor.

The goal of the SCS Honors Undergraduate Research Thesis Program is to introduce students to the breadth of tasks involved in independent research, including library work, problem formulation, experimentation, analysis, technical writing and public speaking. In particular, students write a paper summarizing prior results and current progress in their desired area of research, present a public poster session in December of their senior year describing their current progress, present their final results with a poster and an oral presentation in the year-end university-wide Undergraduate Research Symposium (Meeting of the Minds) and submit a written thesis at the end of their senior year. Students work closely with faculty research advisors to plan and carry out their research. The SCS Honors Undergraduate Research Thesis (07-599) typically starts in the fall semester of the senior year, and spans the entire senior year. Students receive a total of 36 units of academic credit for the thesis work, 18 units per semester. Students should prepare their research prospectus (i.e. proposal of work) during the spring semester of their junior year, and students in this program are advised to plan their schedules carefully to ensure there is ample time to perform the required research for the thesis during the senior year.

Students interested in research are urged to consult with their undergraduate advisor and the SCS Associate Dean no later than the end of their sophomore year in order to plan their workload effectively. Although there is no specific GPA requirement to participate, students are expected to have at least a 3.5 GPA in the core SCS topics relevant to their proposed research to be successful in their work. For those students with no background in research, they may consider using Research and Innovation in Computer Science (07-300, 9 units) as an introduction to the research process in their junior year since this course will introduce students to various research projects going on in the School of Computer Science and important skills that are needed to be an effective researcher. This course leads to a subsequent research practicum (07-400) that allows students to complete a small-scale research study or experiment and present a research poster. Students who use 07-400 to start their senior thesis can use these units toward the required 36 units. Students should consult with their academic advisor concerning how the units earned toward the senior thesis can be used toward elective requirements for their major.

Interested juniors should submit a project prospectus of no more than three pages by the end of their junior year, although submissions over the summer prior to the senior year will also be considered for review. A prospectus must include:

- The name of the research advisor (an SCS faculty member)
- A short abstract (two paragraphs, max)
- A description of the problem to be worked on and its significance
- A tactical description of the proposed research plan, including:
  - a description of the background reading to be carried out,
  - a description of the research contribution,
  - a description of the expected results of the research, and
  - a reasonably detailed timeline for the thesis work
- A bibliography of related work (all references belong here)
- The signature of the research advisor, signifying endorsement of the project and willingness to supervise and evaluate it (or an email confirmation from the research advisor)

Students who need help finding potential advisors should get in touch with their academic advisor or the Associate Dean for Undergraduate Education. Applications to the program are due by the start of the senior year, although submission of applications in the junior year is encouraged. Students completing an outstanding senior thesis based on the judgement of the SCS Undergraduate Review Committee will earn SCS College Honors and can compete for various SCS research awards given out during commencement.

Faculty

UMUT ACAR, Associate Professor, Computer Science Department – Ph.D., Carnegie Mellon University; Carnegie Mellon, 2012–

ANIL ADA, Associate Teaching Professor, Carnegie Mellon University - Ph.D., McGill University; Carnegie Mellon, 2014–

HENRY ADMONI, Assistant Professor, Robotics Institute – Ph.D., Yale University; Carnegie Mellon, 2017–

YUVRAJ AGARWAL, Associate Professor, Institute for Software Research – Ph.D., University of California, San Diego; Carnegie Mellon, 2013–

JONATHAN ALDRICH, Professor, Institute for Software Research – Ph.D., University Of Washington; Carnegie Mellon, 2003–

VINCENT ALEVEN, Professor, Human-Computer Interaction Institute – Ph.D., University Of Pittsburgh; Carnegie Mellon, 2000–

DAVID ANDERSEN, Professor, Computer Science Department – Ph.D., Massachusetts Institute Of Technology; Carnegie Mellon, 2005–

JOHN ANDERSON, R.K. Mellon University Professor – Ph.D., Stanford University; Carnegie Mellon, 1978–

DIMITrios APOSTOLOPOULOS, Senior Systems Scientist, Robotics Institute – Ph.D., Carnegie Mellon University; Carnegie Mellon, 1989–

CHRISTOPHER ATKESON, Professor, Robotics Institute – Ph.D., Massachusetts Institute Of Technology; Carnegie Mellon, 2000–

JAMES AUSOCKI, Associate Professor, Robotics Institute – Ph.D., Carnegie Mellon University; Carnegie Mellon, 2004–

MARIA FLORINA BALCAN, Professor, Machine Learning Department – Ph.D., Carnegie Mellon University; Carnegie Mellon, 2014–

STEPHANIE BALZER, Assistant Research Professor, Computer Science Department – Ph.D., ETH Zurich; Carnegie Mellon, 2016–

ZIV BAR-JOSEPH, Professor, Computational Biology Department – Ph.D., Massachusetts Institute Of Technology; Carnegie Mellon, 2003–

MATTHEW BASS, Assistant Teaching Professor, Institute for Software Research – M.S., Carnegie Mellon University; Carnegie Mellon, 2012–

LUJO BAUER, Professor, Institute for Software Research - Ph.D., Princeton University; Carnegie Mellon, 2015–

NATHAN BECKMANN, Assistant Professor, Computer Science Department – Ph.D., Massachusetts Institute Of Technology; Carnegie Mellon, 2017–

TAYLOR BERG-KIRKPATRICK, Assistant Professor, Language Technologies Institute – Ph.D., University of California at Berkeley; Carnegie Mellon, 2016–

KAREN BERNTSEN, Associate Teaching Professor, Human Computer Interaction Institute - M.S., Duquesne University; Carnegie Mellon, 2005–

JEFFREY BIGHAM, Associate Professor, Human-Computer Interaction Institute - Ph.D., University of Washington; Carnegie Mellon, 2013–

YONATAN BISK, Assistant Professor, Language Technologies Institute – Ph.D., University of Illinois, Urbana- Champaign; Carnegie Mellon, 2020–

ALAN BLACK, Professor, Language Technologies Institute – Ph.D., University Of Edinburgh; Carnegie Mellon, 1999–

GUY BLELLOCH, Professor, Computer Science Department – Ph.D., Massachusetts Institute Of Technology; Carnegie Mellon, 1988–

CHRISTOPHER BOGART, Systems Scientist, Institute for Software research – Ph.D., Oregon State University; Carnegie Mellon, 2017–

DAVID BOURNE, Principal Systems Scientist, Robotics Institute – M.S., University Of Pennsylvania; Carnegie Mellon, 1980–

DAVIN BOYARSKI, Professor – M.F.A., Indiana University; Carnegie Mellon, 1982–
CHRISTOPHER HARRISON, Associate Professor, Human-Computer Interaction Institute – Ph.D., Carnegie Mellon University; Carnegie Mellon, 2014–

ALEXANDER HAUPTMANN, Research Professor, Language Technologies Institute – Ph.D., Carnegie Mellon University; Carnegie Mellon, 1994–

MARTIAL HEBERT, Dean of the School of Computer Science and Professor, Robotics Institute – Ph.D., Paris-XI; Carnegie Mellon, 1984–

DAVID HELD, Assistant Professor, Robotics Institute – Ph.D., Stanford University; Carnegie Mellon, 2017–

JAMES HERBSLEB, Director, Professor, Institute for Software Research – Ph.D., University Of Nebraska; Carnegie Mellon, 2002–

MARJIN HEULE, Associate Professor, Computer Science Department – Ph.D., Delft University of Technology (Netherlands); Carnegie Mellon, 2019–

LEE HILLMAN, Executive Director of MCHI, Human-Computer Interaction Institute – M.S., Carnegie Mellon University; Carnegie Mellon, 2017–

MICHAEL HILTON, Associate Teaching Professor, Institute for Software Research – Ph.D., Oregon State University; Carnegie Mellon, 2017–

JESSICA HODGINS, Professor, Robotics Institute – Ph.D., Carnegie Mellon University; Carnegie Mellon, 2001–

JAN HOFFMANN, Associate Professor, Computer Science Department – Ph.D., Ludwig-Maximilians-Universität and TU Munich; Carnegie Mellon, 2015–

JASON HONG, Associate Professor, Human-Computer Interaction Institute – Ph.D., University Of California at Berkeley; Carnegie Mellon, 2004–

EDUARD HOVY, Research Professor, Language Technologies Institute – Ph.D., Yale University; Carnegie Mellon, 2012–

DANIEL HUBER, Senior Systems Scientist, Robotics Institute – Ph.D., Carnegie Mellon University; Carnegie Mellon, 2002–

SCOTT HUDSON, Professor, Human-Computer Interaction Institute – Ph.D., University Of Colorado; Carnegie Mellon, 1997–

FARNAM JAHANIAN, President, Carnegie Mellon University, and Professor, Computer Science Department – Ph.D., University of Texas at Austin; Carnegie Mellon, 2014–

AAYUSH JAIN, Assistant Professor, Computer Science Department – Ph.D, University of California, Los Angeles; Carnegie Mellon, 2021–

LASZLO JENI, Systems Scientist, Robotics Institute – Ph.D., University of Tokyo; Carnegie Mellon, 2018–

MICHAEL KAESS, Associate Professor – Ph.D., Georgia Institute of Technology; Carnegie Mellon, 2013–

TAKEO KANADE, University Professor, Robotics Institute – Ph.D., Kyoto University; Carnegie Mellon, 1980–

EUNSUK KANG, Assistant Professor, Institute for Software Research – Ph.D., Massachusetts Institute of Technology; Carnegie Mellon, 2017–

JOSHUA KANGAS, Assistant Teaching Professor, Computational Biology Department – PhD, Carnegie Mellon University; Carnegie Mellon, 2018–

GEORGE KANTOR, Research Professor, Robotics Institute – Ph.D., University of Maryland; Carnegie Mellon, 2002–

CHRISTIAN KASTNER, Associate Professor, Institute for Software Research – Ph.D., University of Magdeburg; Carnegie Mellon, 2012–

GEOFF KAUFMAN, Associate Professor, Human Computer Interaction Institute – Ph.D., Ohio State University; Carnegie Mellon, 2015–

DILSUN KAYNUR, Associate Teaching Professor, Computer Science Department – Ph.D., University of Edinburgh; Carnegie Mellon, 2012–

ALONZO KELLY, Professor, Robotics Institute – Ph.D., Carnegie Mellon University; Carnegie Mellon, 1998–

SEUNGJIN KIM, Systems Scientist, Human-Computer Interaction Institute – Ph.D., Gwangju Institute of Science and Technology; Carnegie Mellon, 2011–

SEYOUNG KIM, Associate Professor, Computational Biology Department – Ph.D., University of California at Irvine; Carnegie Mellon, 2010–

CARL KINGSFORD, Professor, Computational Biology Department – Ph.D., Princeton University; Carnegie Mellon, 2012–

KRIS KITANI, Associate Research Professor, Robotics Institute – Ph.D., University of Tokyo; Carnegie Mellon, 2012–

ANIKET KITTUR, Professor, Human-Computer Interaction Institute – Ph.D., University of California At Los Angeles; Carnegie Mellon, 2009–

KENNETH KOEDINGER, Professor, Human-Computer Interaction Institute – Ph.D., Carnegie Mellon University; Carnegie Mellon, 1991–

J. ZICO KOLTER, Associate Professor, Computer Science Department – Ph.D., Stanford University; Carnegie Mellon, 2012–

DAVID KOSBIE, Associate Teaching Professor, Computer Science Department – M.S., Carnegie Mellon University; Carnegie Mellon, 2009–

PRAVESH KOTHARI, Assistant Professor, Computer Science Department – Ph.D., University of Texas at Austin; Carnegie Mellon, 2018–

JOANNIS KOUTIS, Adjunct Assistant Professor, Computer Science Department – Ph.D., Carnegie Mellon University; Carnegie Mellon, 2008–

ROBERT KRAUT, Professor Emeritus, Human-Computer Interaction Institute – Ph.D., Yale University; Carnegie Mellon, 1993–

OLIVER KROEMER, Assistant Professor, Robotics Institute – Ph.D., Technische Universität Darmstadt; Carnegie Mellon, 2017–

CHINMAY KULKARNI, Associate Professor, Human Computer Interaction Institute – Ph.D., Stanford University; Carnegie Mellon, 2015–

CHRISTOPHER LANGMEAD, Associate Professor, Computational Biology Department – Ph.D., Dartmouth University; Carnegie Mellon, 2004–

CLAIRE LE GOUES, Associate Professor, Institute for Software Research – Ph.D., University of Virginia; Carnegie Mellon, 2013–

CHRISTIAN LEBIÈRE, Research Psychologist, Psychology – Ph.D., Carnegie Mellon University; Carnegie Mellon, 1999–

EUN SUN LEE, Associate Teaching Professor, Institute for Software Research – M.S., Carnegie Mellon University; Carnegie Mellon, 2014–

TAI-SING LEE, Professor, Computer Science Department – Ph.D., Massachusetts Institute of Technology; Carnegie Mellon, 1996–

LORRAINE LEVIN, Research Professor, Language Technologies Institute – Ph.D., Massachusetts Institute Of Technology; Carnegie Mellon, 1989–

YUANZHI LI, Assistant Professor, Machine Learning Department – Ph.D., Princeton University; Carnegie Mellon, 2019–

MAXIM LIKACHEV, Associate Research Professor, Robotics Institute – Ph.D., Carnegie Mellon University; Carnegie Mellon, 2010–

CHANGLIU LIU, Assistant Professor, Robotics Institute – Ph.D., University of California, Berkeley; Carnegie Mellon, 2019–

SIMON LUCEY, Associate Research Professor, Robotics Institute – Ph.D., University of Southern Queensland; Carnegie Mellon, 2002–

JIAN MA, Professor, Computational Biology Department – Ph.D., Pennsylvania State University ; Carnegie Mellon, 2016–

JOHN MACKAY, Teaching Professor, Computer Science Department and Mathematics Department – Ph.D., University of Hawaii; Carnegie Mellon, 2003–

NIKOLAS MARTELARO, Assistant Professor, Human Computer Interaction Institute – Ph-D, Stanford University; Carnegie Mellon, 2020–

RUBEN MARTINS, Assistant Research Professor, Institute for Software Research – Ph.D, Technical university of Lisbon; Carnegie Mellon, 2018–

MATTHEW MASON, Professor Emeritus, Computer Science Department – Ph.D, Massachusetts Institute of Technology; Carnegie Mellon, 1982–

ROY MAXION, Research Professor, Computer Science Department – Ph.D., University Of Colorado; Carnegie Mellon, 1984–

JAMES MCCANN, Assistant Professor, Robotics Institute – Ph.D., Carnegie Mellon University; Carnegie Mellon, 2017–

BRUCE McLAREN, Associate Research Professor, Human-Computer Interaction Institute – Ph.D., University Of Pittsburgh; Carnegie Mellon, 2003–

FLORIAN METZE, Associate Research Professor, Language Technologies Institute – Ph.D., Universität Karlsruhe; Carnegie Mellon, 2009–

NATHAN MICHAEL, Assistant Research Professor, Robotics Institute – Ph.D., University of Pennsylvania; Carnegie Mellon, 2012–

GARY MILLER, Professor Emeritus, Computer Science Department – Ph.D., University of California, Berkeley; Carnegie Mellon, 1988–

HEATHER MILLER, Assistant Professor, Institute for Software Research – Ph.D., École Polytechnique Fédérale de Lausanne; Carnegie Mellon, 2018–

EDUARDO MIRANDA, Associate Teaching Professor, Institute for Software Research – M.S./M.Eng., University of Linköping/University of Ottawa; Carnegie Mellon, 2006–

TERUKO MITAMURA, Research Professor, Language Technologies Institute – Ph.D., University Of Pittsburgh; Carnegie Mellon, 1990–
TOM MITCHELL, University Professor, Machine Learning Department – Ph.D., Stanford University; Carnegie Mellon, 1986–

STEFAN MITSCH, Systems Scientist, Computer Science Department – Ph.D., Johannes Kepler University; Carnegie Mellon, 2016–

HOSEIN MOHIMANI, Assistant Professor, Computational Biology Department – Ph.D., University of California, San Diego; Carnegie Mellon, 2017–

ALAN MONTGOMERY, Associate Professor of Marketing – Ph.D., University Of Chicago; Carnegie Mellon, 1999–

IGOR MORDATCH, Assistant Professor, Robotics Institute – Ph.D., University of Washington; Carnegie Mellon, 2017–

LOUIS-PHILIPPE MORENCY, Associate Professor, Language Technologies Institute – Ph.D., Massachusetts Institute of Technology; Carnegie Mellon, 2015–

DOMINIK MORITZ, Assistant Professor, Robotics Institute – Ph.D. University of Washington; Carnegie Mellon, 2020–

JAMES MORRIS, Professor, Emeritus, Human-Computer Interaction Institute – Ph.D., Massachusetts Institute Of Technology; Carnegie Mellon, 1982–

DAVID MORTENSEN, Research Scientist, Language Technologies Institute – Ph.D. University of California, Berkeley; Carnegie Mellon, 2015–

JACK MOSTOW, Research Professor Emeritus, Robotics Institute – Ph.D., Carnegie Mellon University; Carnegie Mellon, 1992–

TODD MOWRY, Professor, Computer Science Department – Ph.D., Stanford University; Carnegie Mellon, 1997–


ROBERT MURPHY, Professor Emeritus, Computational Biology Department – Ph.D., California Institute Of Technology; Carnegie Mellon, 1983–

BRAD MYERS, Professor, Human-Computer Interaction Institute – Ph.D., University Of Toronto; Carnegie Mellon, 1987–

PRIYA NARASIMHAN, Professor – Ph.D., University Of California; Carnegie Mellon, 2001–

SRINIVASA NARASIMHAN, Professor, Interim Director, Robotics Institute – Ph.D., Columbia University; Carnegie Mellon, 2004–

GRAHAM NEUBIG, Associate Professor, Language Technologies Institute – Ph.D., Kyoto University; Carnegie Mellon, 2016–

CHRISTINE NEUWIRTH, Professor – Ph.D., Carnegie Mellon University; Carnegie Mellon, 2004–

ILLAH NOURBakhsh, Professor, Robotics Institute – Ph.D., Stanford University; Carnegie Mellon, 1997–

ERIC NYBERG, Professor, Language Technologies Institute – Ph.D., Carnegie Mellon University; Carnegie Mellon, 1989–

RYAN O'DONNELL, Professor, Computer Science Department – Ph.D., Massachusetts Institute Of Technology; Carnegie Mellon, 2006–

KEMAL OFLAZER, Associate Dean of Research, Language Technologies Institute – Ph.D., Carnegie Mellon University; Carnegie Mellon, 2008–

AMY OGAN, Associate Professor, Human-Computer Interaction Institute – Ph.D., Carnegie Mellon University; Carnegie Mellon, 2014–

DAVID O’HALLARON, Professor, Computer Science Department – Ph.D., University of Virginia; Carnegie Mellon, 1989–

JEAN OH, Associate Research Professor, Robotics Institute – Ph.D., Carnegie Mellon University; Carnegie Mellon, 2019–

IRVING OPPENHEIM, Professor – Ph.D., University of Cambridge; Carnegie Mellon, 1973–

MATTHEW O’TOOLE, Assistant Professor, Robotics Institute and Computer Science Department – Ph.D., University of Toronto; Carnegie Mellon, 2018–

PATRICK PARK, Assistant Professor, Institute for Software Research – Ph.D., Cornell University; Carnegie Mellon, 2021–

BRYAN PARNO, Associate Professor – Ph.D., Carnegie Mellon University; Carnegie Mellon, 2017–

DEEPAK PATHAK, Assistant Professor, Robotics Institute – Ph.D., University of California, Berkeley; Carnegie Mellon, 2020–

ANDREW PAVLO, Associate Professor, Computer Science Department – Ph.D., Brown University; Carnegie Mellon, 2013–

ADAM PERER, Assistant Research Professor, Human Computer Interaction Institute – Ph.D., University of Maryland; Carnegie Mellon, 2018–

JUergen Pfeffer, Assistant Research Professor, Institute for Software Research – Ph.D., Vienna University of Technology; Carnegie Mellon, 2012–

Andreas Pfennig, Assistant Professor, Computational Biology Department – Ph.D., Duke University; Carnegie Mellon, 2015–

Frank Pfennig, Professor, Computer Science Department – Ph.D., Carnegie Mellon University; Carnegie Mellon, 1986–

Andre Platzer, Professor, Computer Science Department – Ph.D., University of Oldenburg; Carnegie Mellon, 2008–

Barnabas Poczos, Associate Professor, Machine Learning Department – Ph.D., Eotvos Lorand University; Carnegie Mellon, 2012–

Nancy Pollard, Professor, Robotics Institute – Ph.D., Massachusetts Institute Of Technology; Carnegie Mellon, 2002–

Brian Railing, Assistant Teaching Professor, Computer Science Department – Ph.D., Georgia Institute of Technology; Carnegie Mellon, 2016–

Bhiksha Raj Ramakrishnan, Professor, Language Technologies Institute – Ph.D., Carnegie Mellon University; Carnegie Mellon, 2008–

Deva Ramanan, Professor, Robotics Institute – Ph.D., University of California at Berkeley; Carnegie Mellon, 2015–

Pradeep Ravikumar, Associate Professor, Machine Learning Department – Ph.D., Carnegie Mellon University; Carnegie Mellon, 2016–

Raj Reddy, University Professor, Institute for Software Research – Ph.D., Stanford University; Carnegie Mellon, 1969–

Andréj Risteski, Assistant Professor, Machine Learning Department – Ph.D., Princeton University; Carnegie Mellon, 2019–

Kelly Rivers, Assistant Teaching Professor, Computer Science Department – Ph.D., Carnegie Mellon University; Carnegie Mellon, 2017–

Cameron Riviere, Research Professor, Robotics Institute – Ph.D., Johns Hopkins University; Carnegie Mellon, 1995–

David Root, Associate Teaching Professor, Institute for Software Research – M.P.M., Carnegie Mellon University; Carnegie Mellon, 2002–

Carolyn Rose, Professor, Language Technologies Institute – Ph.D., Carnegie Mellon University; Carnegie Mellon, 2003–

Ronald Rosenfeld, Professor and Department Head, Machine Learning Department – Ph.D., Carnegie Mellon University; Carnegie Mellon, 1995–

Stéphanie Rosenthal, Assistant Teaching Professor, Computer Science Department – Ph.D., Carnegie Mellon University; Carnegie Mellon, 2019–

Steven Rudich, Professor, Computer Science Department – Ph.D., University of California; Carnegie Mellon, 1989–


Majd Sakr, Teaching Professor, Computer Science Department – Ph.D., University of Pittsburgh; Carnegie Mellon, 2006–

Ruslan Salakhutdinov, Professor, Machine Learning Department – Ph.D., University of Toronto; Carnegie Mellon, 2016–

Tuomas Sandholm, Professor, Computer Science Department – Ph.D., University of Massachusetts; Carnegie Mellon, 2001–

Mahadev Satyanarayanan, Professor, Computer Science Department – Ph.D., Carnegie Mellon University; Carnegie Mellon, 1983–

Richard Scheines, Dean, Dietrich College and Professor, Philosophy – Ph.D., University of Pittsburgh; Carnegie Mellon, 1988–

Sebastian Scherer, Associate Research Professor, Robotics Institute – Ph.D., Carnegie Mellon University; Carnegie Mellon, 2010–

William Scherlis, Professor and Director, Institute for Software Research – Ph.D., Stanford University; Carnegie Mellon, 1989–

Bradley Schmerl, Principal Systems Scientist, Computer Science Department – Ph.D., Flinders University of South Australia; Carnegie Mellon, 2000–

Jeff Schneider, Research Professor, Robotics Institute – Ph.D., University of Rochester; Carnegie Mellon, 1995–

Russell Schwartz, Professor of Biological Sciences and Department Head, Computational Biology Department – Ph.D, Massachusetts Institute of Technology; Carnegie Mellon, 2002–
DANA SCOTT, Professor Emeritus, Computer Science Department – Ph.D., Princeton University; Carnegie Mellon, 1981-

TEDDY SEIDENFELD, Herbert A. Simon Professor - Ph.D., Columbia University; Carnegie Mellon, 1985-

SRINIVASAN SESHAN, Professor and Department Head, Computer Science Department – Ph.D., University of California; Carnegie Mellon, 2000-

NIHAR SHAH, Assistant Professor, Machine Learning Department – Ph.D., University of California at Berkeley; Carnegie Mellon, 2017-

MAYUR SHAMOS, Teaching Professor, Language Technologies Institute and Institute for Software Research – Ph.D., Yale University; Carnegie Mellon, 1975-

MARY SHAW, University Professor, Institute for Software Research – Ph.D., Carnegie Mellon University; Carnegie Mellon, 1965-

SKIP SHELLY, Associate Teaching Professor, Human Computer Interaction Institute - B.F.A., Carnegie Mellon University; Carnegie Mellon, 2017-

JUSTINE SHERRY, Assistant Professor, Computer Science Department – Ph.D., University of California at Berkeley; Carnegie Mellon, 2017-

HIROKAZU SHIRADO, Assistant Professor, Human Computer Interaction Institute – Ph.D., Yale University; Carnegie Mellon, 2019-

DOUGLAS SICKER, Professor, Institute for Software Research – Ph.D., University of Pittsburgh; Carnegie Mellon, 2014-

MEL SIEGEL, Associate Research Professor Emeritus, Robotics Institute – Ph.D., University of Colorado; Carnegie Mellon, 1962-

DANIEL SIEWIAREK, University Professor, Human-Computer Interaction Institute – Ph.D., Stanford University; Carnegie Mellon, 1972-

REID SIMMONS, Research Professor, Robotics Institute – Ph.D., Massachusetts Institute of Technology; Carnegie Mellon, 1988-

AARTI SINGH, Associate Professor, Machine Learning Department - Ph.D., University of Wisconsin At Madison; Carnegie Mellon, 2009-

RITA SINGH, Associate Research Professor, Language Technologies Institute – Ph.D., National Geophysical Research Institute; Carnegie Mellon, 2010-

DANIEL SIEGEL, Professor, Computer Science Department – Ph.D., Stanford University; Carnegie Mellon, 1985-

STEPHEN SMITH, Research Professor, Robotics Institute – Ph.D., University of Pittsburgh; Carnegie Mellon, 1982-

PETER SPIRTES, Professor, Philosophy – Ph.D., University of Pittsburgh; Carnegie Mellon, 1983-

JOHN STAMPER, Associate Professor, Human-Computer Interaction Institute – Ph.D., University of North Carolina At Charlotte; Carnegie Mellon, 2009-

PETER STEENKISTE, Professor, Computer Science Department – Ph.D., Stanford University; Carnegie Mellon, 1987-

MARK STEHLIK, Teaching Professor, Computer Science Department – B.S., Pace University; Carnegie Mellon, 1981-

AARON STEINFELD, Associate Research Professor, Robotics Institute – Ph.D., University of Michigan; Carnegie Mellon, 2001-

GEORGE STETTEN, Adjunct Research Professor, Robotics Institute – Ph.D., University of North Carolina; Carnegie Mellon, 1999-

EMMA STRUBLER, Assistant Professor, Language Technologies Institute – Ph.D., University of Massachusetts, Amherst; Carnegie Mellon, 2020-

JOSHUA SUNSHINE, Systems Scientist, Institute for Software Research – Ph.D., Carnegie Mellon University; Carnegie Mellon, 2014-

KLAUS SUTNER, Teaching Professor, Computer Science – Ph.D., University of Munich; Carnegie Mellon, 1995-

KATIA SYCARA, Research Professor, Robotics Institute – Ph.D., Georgia Institute of Technology; Carnegie Mellon, 1987-

WENNIE TABIB, Systems Scientist, Robotics Institute – Ph.D., Carnegie Mellon University; Carnegie Mellon, 1987-

AMEET TALWALKAR, Assistant Professor, Machine Learning Department – Ph.D., New York University, Courant Institute; Carnegie Mellon, 2017-

ZEYNEP TEMEL, Assistant Professor, Robotics Institute – Ph.D., Sabanci University (Istanbul, Turkey); Carnegie Mellon, 2019-

RYAN TIBSHIRANI, Professor, Machine Learning Department – Ph.D, Stanford University; Carnegie Mellon, 2011-

DAVID TOURETZKY, Research Professor, Computer Science Department – Ph.D., Carnegie Mellon University; Carnegie Mellon, 1984-

MATTHEW TRAVERS, Systems Scientist, Robotics Institute – Ph.D., Northwestern University; Carnegie Mellon, 2013-

YULIA TSVETKOVA, Assistant Professor, Language Technologies Institute – Ph.D., Carnegie Mellon University; Carnegie Mellon, 2017-

BOGDAN VASILESCU, Assistant Professor, Institute for Software Research – Ph.D., Eindhoven University of Technology; Carnegie Mellon, 2016-

MANUELA VELOSO, Professor Emeritus, Machine Learning Department – Ph.D. Carnegie Mellon University; Carnegie Mellon, 1992-

RASHMI VINYAK, Assistant Professor, Computer Science Department – Ph.D., University of California at Berkeley; Carnegie Mellon, 2017-

PAT VIRTUE, Assistant Teaching Professor, Computer Science Department and Machine Learning Department – Ph.D., University of California at Berkeley; Carnegie Mellon, 2018-

ALEXANDER WAIBEL, Professor, Language Technologies Institute - Ph.D., Carnegie Mellon University; Carnegie Mellon, 1988-

WEINA WANG, Assistant Professor, Computer Science Department – Ph.D., Arizona State University; Carnegie Mellon, 2018-

LEILA WHEBE, Assistant Professor, Machine Learning Department – Ph.D., Carnegie Mellon University; Carnegie Mellon, 2018-

DAVID WETTERGREEN, Research Professor, Robotics Institute – Ph.D., Carnegie Mellon University; Carnegie Mellon, 2000-

WILLIAM RED WHITTAKER, University Research Professor, Robotics Institute - Ph.D., Carnegie Mellon University; Carnegie Mellon, 1973-

WEI WU, Senior Systems Scientist, Computational Biology Department – Ph.D., Rutgers University; Carnegie Mellon, 2011-

FRANCESKA XHAKAJ, Assistant Teaching Professor, Computer Science Department/ Human Computer Interaction Institute – Ph.D. Carnegie Mellon University; Carnegie Mellon, 2021-

POE ERIC XING, Professor, Machine Learning Department – Ph.D., University Of California At Berkeley; Carnegie Mellon, 2004-

MIN XU, Assistant Professor, Computational Biology Department – Ph.D., University of Southern California; Carnegie Mellon, 2016-

YIMING YANG, Professor, Language Technologies Institute – Ph.D., Kyoto University; Carnegie Mellon, 1996-

LINING YAO, Assistant Professor, Human Computer Interaction Institute – Ph.D., Massachusetts Institute of Technology; Carnegie Mellon, 2017-

WENZHENG YUAN, Assistant Professor, Robotics Institute – Ph.D., Massachusetts Institute of Technology; Carnegie Mellon, 1992-

HAIYI ZHU, Assistant Professor, Human Computer Interaction Institute – Ph.D., Carnegie Mellon University; Carnegie Mellon, 2019-

JUN-YAN ZHU, Assistant Professor, Robotics Institute – Ph.D., University of California; Berkeley; Carnegie Mellon, 2020-

JOHN ZIMMERMANN, Professor, Human-Computer Interaction Institute – M.Des., Carnegie Mellon University; Carnegie Mellon, 2002-