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 Computational Biology Department, the Human-Computer Interaction Institute, the Institute for Software Research, the Language Technologies Institute, the Machine Learning Department, and the Robotics Institute. Together, these units make SCS a world leader in research and education. Over the last five years, SCS has launched three new primary undergraduate majors in Computational Biology, Artificial Intelligence (the first of its kind in the United States), and 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 Artificial Intelligence
- Additional major in Computational Biology
- Additional major in Computer Science
- Additional major in Human-Computer Interaction (Interdisciplinary)
- Additional major in Robotics
- Minor in Artificial Intelligence
- Minor in Computational Biology
- Minor in Computer Science
- Minor in Human-Computer Interaction
- Minor in Information Security, Privacy and Policy
- 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, additional major, minor)
- 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/undergraduatecomputerscience/) (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/additionalmajors/minors/)

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 Computation and Human-Computer Interaction. Students with artistic and computing interests may be given the option to pursue a major in 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 Human-Computer Interaction. Suitable 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/aboutcmu/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:
• A first year student will be suspended if the QPA from each semester is below 1.75,
• 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 submitting this form to their academic advisor, and
2. submitting an additional written statement to their academic advisor and 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 to the academic advisor and the SCS Associate Dean for Undergraduate Programs, 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 their student job and 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 subsequent semesters after 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 business days of notification if, under consultation with their academic advisor, they feel that the decision was made in error and they have additional 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 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, for students taking a leave for academic reasons, 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 on-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.

Transfer into SCS / Dual-Degree

Undergraduate students admitted to colleges at CMU other than SCS and wishing to transfer to SCS or pursue a dual degree in SCS should consult with the Director or Program Coordinator of the desired SCS major during their first year. See the individual program pages for the names of the current directors and program coordinators, along with their contact information.

- For the Artificial Intelligence primary major, students must complete 15-122, 15-150, 15-281, 10-315, one of 15-210, 15-285, 15-286, 36-225, 21-325, 36-218 or 15-259, 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 primary major, students must complete 21-127 (or equivalent), 15-122, 15-251, 15-351 (or 15-210), 03-121 and either 02-251 or 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.)
- 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 Human-Computer Interaction primary major, students must complete 21-127 (or equivalent), 15-122, 15-150, 05-410, one of 15-210 or 15-213, and one of 05-470 or 05-651, 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 may apply for transfer by the start of 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 coursework.
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.

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. Consult the website for the individual SCS major for more information about expected courses to take to demonstrate interest and ability.
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 (http://coursecatalog.web.cmu.edu/schools-colleges/ schoolofcomputerscience/undergraduatecomputationalbiology/) program page 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 Name</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>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>09-218</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>10-120</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>Mechanics I: 2D Design</td>
<td>10</td>
</tr>
<tr>
<td>24-291-24-381</td>
<td>Environmental Systems on a Changing Planet - Environmental Systems on a Changing Planet: Science Engineering Addendum (must take both courses together)</td>
<td>12</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>12</td>
</tr>
<tr>
<td>33-151</td>
<td>Matter and Interactions I</td>
<td>12</td>
</tr>
<tr>
<td>33-142</td>
<td>Physics II for Engineering and Physics Students</td>
<td>12</td>
</tr>
<tr>
<td>or 33-152</td>
<td>Matter and Interactions II</td>
<td>12</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:

- Quantitative Cell and Molecular Biology Laboratory (can be paired with a course in Biology 03-xxx for two courses in one department)
- Modern Biology Laboratory (This 3 unit lab together with 09-105 satisfies the lab requirement.)

The following MCS and CIT courses cannot be used to satisfy the Science and Engineering requirement (see note below this list for additional exceptions and conditions):

- Computational Molecular Biology and Genomics | 9 |
- Computational Methods for Biological Modeling and Simulation | 9 |
- Fundamentals of Software Development and Problem Solving | 12 |
- Mathematical Methods of Chemical Engineering | 12 |
- Atoms, Molecules and Chemical Change | 9 |
- The Illusion and Magic of Food | 6 |
- Kitchen Chemistry Sessions | 3 |
- The Design and Making of Skin and Hair Products | 3 |
- Basics of Food Science | 3 |
- Professional Communication Skills in Chemistry | 3 |
- Kitchen Chemistry Sessions | 3 |
- Mathematical Methods for Chemists | 9 |
- Introduction to Professional Writing in CEE | 9 |
- Computation and Data Science for Civil & Environmental Engineering | 9 |
- Twisted Signals: Multimedia Processing for the Arts | 10 |
- ECE Sophomore Seminar | 1 |
- Mathematical Foundations of Electrical Engineering | 12 |
- Introduction to Computer Systems | 12 |
- Introduction to Computer Security | 12 |
- Network Security | 12 |
- Secure Software Systems | 12 |
- Computational Techniques in Engineering | 12 |
- Computer Networks | 12 |
- Introduction to Machine Learning for Engineers | 12 |
- Principles and Engineering Applications of AI | 12 |
- Advanced Probability & Statistics for Engineers | 12 |
- Telecommunications Technology and Policy for the Internet Age | 12 |
- Introduction to Computer Security | 12 |
- Rapid Prototyping of Computer Systems | 12 |
- Introduction to Engineering and Public Policy | 12 |
- Ethics and Policy Issues in Computing (or 17-200) | 9 |
- The American Railroad: Decline and Renaissance in the Age of Deregulation | 6 |
1. with Category 1A: Cognitive Studies which is a subset of Category Undergraduate Education, if the minis meet the goals of the desired Category 3. Students may use two minis totaling 9 units or more to Complete three courses, one each from Category 1, Category 2, and
B. Breadth Requirement (minimum 27 units: 9 units each)

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/intercollegeprograms/bxaintercollege/ #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:

or two of these three writing minis for 9 units total:

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.

Organizational Behavior 9
Dangerous Ideas in Science and Society 9
Introduction to Ethics 9

Nature of Reason 9
Nature of Language 9
Philosophy of Social Science 9
Problems of Mind and Body: Meaning and Doing 9
Mind and Body: The Objective and the Subjective 9
Metaphysics 9
Ethical Theory 9
Introduction to Psychology 9
Psychopathology 9
Cognitive Psychology 9
Human Information Processing and Artificial Intelligence 9
Principles of Child Development 9
Social Psychology 9
Personality 9
Psychopathology 9
Perception 9
Visual Cognition 9
Language and Thought 9
Reason, Passion and Cognition 9
Human Intelligence and Human Stupidity 9
Category 1A (for Artificial Intelligence majors): Cognitive Studies - this requirement explores how the brain and the mind work.

Cognitive Psychology 9
Human Information Processing and Artificial Intelligence 9

Perception 9
Visual Cognition 9
Language and Thought 9

Sampling, Survey and Society 9
Topics of Law: Introduction to Intellectual Property Law 9
Business, Society and Ethics 9
Business Law 9
Principles of Microeconomics 9
Principles of Macroeconomics 9
Making History: How to Think About the Past (and Present) 9
History of Democracy: Thinking Beyond the Self 9
Women in American History 9
Introduction to the History of Science 9
History of American Public Policy 9
U. S. Business History: 1870 to the Present 9
Women, Politics, and Protest 9
Documenting Human Rights 9
Body Politics: Women and Health in America 9
The History of Capitalism 9
Introduction to Political Philosophy 9
Social Structure, Public Policy & Ethics 9
Environmental Ethics 9
Medical Ethics 9
Philosophy of Economics 9
Social and Political Philosophy 9
Decision Processes in American Political Institutions 9
Foundations of Political Economy 9
Comparative Politics 9
International Political Economy 9
Nonviolent Conflict and Revolution 9
The Future of Democracy 9
Theories of International Relations 9
Representation and Voting Rights 9
Diplomacy and Statecraft 9
The Politics of Fake News and Misinformation 9
**C. Humanities and Arts Electives (minimum 27 units)**

Complete 3 non-technical courses of at least 9 units each from any of the departments in the Dietrich College of Humanities & Social Sciences or the College of Fine Arts. Some of the courses taught in these units are considered technical courses and may not be used to satisfy this requirement (see Deletions below). Additionally, a select set of courses from Business Administration and from Environmental and Public Policy can also count for this requirement (see Additions below). Students may combine humanities/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:

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Course Title</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>51-257</td>
<td>Introduction to Computing for Creative Practices</td>
<td>10</td>
</tr>
<tr>
<td>51-327</td>
<td>Design Center: Introduction to Web Design</td>
<td>9</td>
</tr>
<tr>
<td>51-328</td>
<td>Design Center: Design for Digital Systems</td>
<td>9</td>
</tr>
<tr>
<td>76-388</td>
<td>Coding for Humanities and History</td>
<td>9</td>
</tr>
<tr>
<td>76-481</td>
<td>Introduction to Multimedia Design</td>
<td>12</td>
</tr>
<tr>
<td>76-487</td>
<td>Web Design</td>
<td>12</td>
</tr>
<tr>
<td>80-210</td>
<td>Logic and Proofs</td>
<td>9</td>
</tr>
<tr>
<td>80-211</td>
<td>Logic and Mathematical Inquiry</td>
<td>9</td>
</tr>
<tr>
<td>80-212</td>
<td>Arguments and Logical Analysis</td>
<td>9</td>
</tr>
<tr>
<td>80-305</td>
<td>Decision Theory</td>
<td>9</td>
</tr>
<tr>
<td>80-310</td>
<td>Formal Logic</td>
<td>9</td>
</tr>
<tr>
<td>80-311</td>
<td>Undecidability and Incompleteness</td>
<td>9</td>
</tr>
<tr>
<td>80-314</td>
<td>Causal Discovery, Statistics, and Machine Learning</td>
<td>9</td>
</tr>
<tr>
<td>80-315</td>
<td>Modal Logic</td>
<td>9</td>
</tr>
<tr>
<td>80-316</td>
<td>Logic and AI</td>
<td>9</td>
</tr>
<tr>
<td>80-325</td>
<td>Foundations of Causation and Machine Learning</td>
<td>9</td>
</tr>
<tr>
<td>80-405</td>
<td>Game Theory</td>
<td>9</td>
</tr>
<tr>
<td>80-411</td>
<td>Proof Theory</td>
<td>9</td>
</tr>
<tr>
<td>80-413</td>
<td>Category Theory</td>
<td>9</td>
</tr>
<tr>
<td>80-419</td>
<td>Interactive Theorem Proving</td>
<td>9</td>
</tr>
<tr>
<td>80-514</td>
<td>Categorical Logic</td>
<td>9</td>
</tr>
<tr>
<td>80-521</td>
<td>Seminar on Formal Epistemology: Belief and Evidence</td>
<td>Var.</td>
</tr>
<tr>
<td>85-219</td>
<td>Biological Foundations of Behavior</td>
<td>9</td>
</tr>
<tr>
<td>85-310</td>
<td>Research Methods in Cognitive Psychology</td>
<td>9</td>
</tr>
<tr>
<td>85-314</td>
<td>Cognitive Neuroscience Research Methods</td>
<td>9</td>
</tr>
<tr>
<td>85-414</td>
<td>Cognitive Neuropsychology</td>
<td>9</td>
</tr>
<tr>
<td>85-426</td>
<td>Learning in Humans and Machines</td>
<td>9</td>
</tr>
<tr>
<td>88-251</td>
<td>Empirical Research Methods</td>
<td>9</td>
</tr>
<tr>
<td>88-372</td>
<td>Social and Emotional Brain</td>
<td>9</td>
</tr>
</tbody>
</table>

**Additions**

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

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Course Title</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>82-298</td>
<td>Bilingual &amp; Bicultural Experiences in the US</td>
<td>9</td>
</tr>
<tr>
<td>82-299</td>
<td>Russian Cinema: From the Bolshevik Revolution to Putin's Russia</td>
<td>9</td>
</tr>
<tr>
<td>82-307</td>
<td>19th Century Russian Masterpieces</td>
<td>9</td>
</tr>
<tr>
<td>82-309</td>
<td>French &amp; Francophone Cultures</td>
<td>9</td>
</tr>
<tr>
<td>82-310</td>
<td>Literature of the Arabic-speaking World</td>
<td>9</td>
</tr>
<tr>
<td>82-327</td>
<td>The Emergence of the German Speaking World</td>
<td>9</td>
</tr>
<tr>
<td>82-333</td>
<td>Chinese Language and Culture</td>
<td>9</td>
</tr>
<tr>
<td>82-342</td>
<td>Spain: Language and Culture</td>
<td>9</td>
</tr>
<tr>
<td>82-343</td>
<td>Latin America: Language and Culture</td>
<td>9</td>
</tr>
<tr>
<td>82-344</td>
<td>U.S. Latinos: Language and Culture</td>
<td>9</td>
</tr>
<tr>
<td>82-345</td>
<td>Introduction to Hispanic Literary and Cultural Studies</td>
<td>9</td>
</tr>
<tr>
<td>82-436</td>
<td>Introduction to Classical Chinese</td>
<td>9</td>
</tr>
</tbody>
</table>
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 short 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 07-599 SCS Honors Undergraduate Research Thesis 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 QPA requirement to participate, students are expected to have at least a 3.5 QPA 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 07-300 Research and Innovation in Computer Science (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 Research Practicum in Computer Science (12 units), that allows students to complete a small-scale research study or experiment and present a research poster. Students who use this practicum to start their senior thesis can use the 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 3-4 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–

HENNY 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 ALEVYN, 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–

SWARNALATHA ASHOK, Associate Teaching Professor, Institute for Software Research – MSc(Tech), Birla Institute of Technology and Science; Carnegie Mellon, 2022–

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

JAMES BAGNELL, 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–

MANUEL BLUM, University Professor Emeritus, Computer Science Department – Ph.D., Massachusetts Institute of Technology; Carnegie Mellon, 2001–

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–

DANIEL BOYARSKI, Professor – M.F.A., Indiana University; Carnegie Mellon, 1982–

TRAVIS BREAUX, Associate Professor, Institute for Software Research – Ph.D., North Carolina State University; Carnegie Mellon, 2010–

STEPHEN BROOKES, Professor, Computer Science Department – Ph.D., Oxford University; Carnegie Mellon, 1981–


FRASER BROWN, Assistant Professor, Institute for Software Research – Ph.D., Stanford University; Carnegie Mellon, 2022–

RANDEL BRYANT, University Professor Emeritus, Computer Science Department – Ph.D., Massachusetts Institute of Technology; Carnegie Mellon, 1984–

JAMES CALLAN, Professor and Director, Language Technologies Institute – Ph.D., University Of Massachusetts; Carnegie Mellon, 1999–

JAVIER CAMARA-MORENO, Systems Scientist, Institute for Software Research – Ph.D., University of Malaga; Carnegie Mellon, 2015–

OANA CARJA, Assistant Professor, Computational Biology – Ph.D., Stanford University; Carnegie Mellon, 2019–

KATHLEEN CARLEY, Professor, Institute for Software Research – Ph.D., Harvard University; Carnegie Mellon, 1984–

JACOBO CARRASQUEL, Associate Teaching Professor Emeritus, Computer Science Department – Ph.D., Carnegie Mellon University; Carnegie Mellon, 1983–

PATRICK CARRINGTON, Assistant Professor, Human Computer Interaction Institute – Ph.D., University of Maryland; Carnegie Mellon, 2019–

JUSTINE CASSELL, Professor, Language Technologies Institute – Ph.D., University of Chicago; Carnegie Mellon, 2010–

ILIANO CERVESATO, Teaching Professor, Computer Science Department – Ph.D., University of Toronto; Carnegie Mellon, 2016–

HENRY CHAI, Assistant Teaching Professor, Machine Learning Department – Ph.D., Washington University, Saint Louis; Carnegie Mellon, 2022–

TIANQI CHEN, Assistant Professor, Machine Learning Department / Computer Science Department – Ph.D., University of Washington; Carnegie Mellon, 2020–

HOWARD CHOSET, Professor, Robotics Institute – Ph.D., California Institute Of Technology; Carnegie Mellon, 1996–

NICOLAS CHRISTIN, Professor, Institute for Software Research – Ph.D., University of Virginia; Carnegie Mellon, 2017–

WILLIAM COHEN, Professor, Machine Learning Department – Ph.D., Rutgers University; Carnegie Mellon, 2003–

PHILLIP COMPEAU, Associate Teaching Professor, Computational Biology Department – Ph.D., University of California, San Diego; Carnegie Mellon, 2015–

VINCENT CONITZER, Professor, Computer Science Department – Ph.D., Carnegie Mellon University; Carnegie Mellon, 2022–

ALBERT CORBETT, Associate Research Professor Emeritus, Human-Computer Interaction Institute – Ph.D., University Of Oregon; Carnegie Mellon, 1983–

THOMAS CORTINA, Associate Dean for Undergraduate Programs and Teaching Professor, Computer Science Department – Ph.D., Polytechnic University (Brooklyn); Carnegie Mellon, 2004–

KEENAN CRANE, Associate Professor, Robotics Institute – Ph.D., California Institute of Technology; Carnegie Mellon, 2015–

LORRIE CRANOR, Professor, Institute for Software Research – Ph.D., Washington University; Carnegie Mellon, 2003–

KARL CRARY, Associate Professor, Computer Science Department – Ph.D., Cornell University; Carnegie Mellon, 1998–


ROGER DANNENBERG, Professor Emeritus, Computer Science Department – Ph.D., Carnegie Mellon University; Carnegie Mellon, 1982–

SAUVIK DAS, Assistant Professor, Human Computer Interaction Institute – Ph.D., Carnegie Mellon University; Carnegie Mellon, 2022–

FERNANDO DE LA TORRE FRADE, Associate Research Professor, Robotics Institute – Ph.D., La Salle School of Engineering; Carnegie Mellon, 2002–

JOHN DOLAN, Principal Systems Scientist, Robotics Institute - Ph.D., Carnegie Mellon University; Carnegie Mellon, 1991–

ARTUR DUBAWSKI, Research Professor, Robotics Institute – Ph.D., Institute of Fundamental Technological Research; Carnegie Mellon, 2003–

DAVID ECKHARDT, Teaching Professor, Computer Science Department – Ph.D., Carnegie Mellon University; Carnegie Mellon, 2003–

WILLIAM EDDY, Professor – Ph.D., Yale University; Carnegie Mellon, 1976–

JEFFREY EPPINGER, Professor Of The Practice, Institute for Software Research – Ph.D., Carnegie Mellon University; Carnegie Mellon, 2001–

MICHAEL ERMANN, Professor, Robotics Institute – Ph.D., Massachusetts Institute Of Technology; Carnegie Mellon, 1989–

ZACKORY ERICKSON, Assistant Professor, Robotics Institute – Ph.D, Georgia Institute of Technology; Carnegie Mellon, 2021–

MAXINE ESKENAZI, Principal Systems Scientist, Language Technologies Institute – Ph.D., University Of Paris; Carnegie Mellon, 1994–

MOTAHHARE ESLAMI, Assistant Professor, Human Computer Interaction Institute – Ph.D, University of Illinois, Urbana- Champaign; Carnegie Mellon, 2020–

SCOTT FAHLMAN, Professor Emeritus, Language Technologies Institute – Ph.D., Massachusetts Institute Of Technology; Carnegie Mellon, 1978–

CHRISTOS FALOUTSOS, Professor, Computer Science Department – Ph.D., University Of Toronto; Carnegie Mellon, 1997–

FEI FANG, Assistant Professor, Institute for Software Research – Ph.D., University of Southern California; Carnegie Mellon, 2017–

JODI FORLIZZI, Professor, Director; Human-Computer Interaction Institute – Ph.D., Carnegie Mellon University; Carnegie Mellon, 2000–

SARAH FOX, Assistant Professor, Human Computer Interaction Institute – Ph.D, University of Washington; Carnegie Mellon, 2020–

KATE FRAGIADAKI, Assistant Professor, Machine Learning Department – Ph.D., University of Pennsylvania ; Carnegie Mellon, 2016–


MATTHEW FREDRIKSON, Associate Professor, Computer Science Department – Ph.D., University of Wisconsin; Carnegie Mellon, 2015–

DANIEL FRIED, Assistant Professor, Language Technologies Institute – Ph.D., University of California at Berkeley; Carnegie Mellon, 2022–

JOHN GALEOTTI, Senior Systems Scientist, Robotics Institute – Ph.D., Carnegie Mellon University; Carnegie Mellon, 2014–

DAVID GARLAN, Professor, Institute for Software Research – Ph.D., Carnegie Mellon University; Carnegie Mellon, 1990–

CHARLES GARROD, Associate Teaching Professor, Institute for Software Research – Ph.D., Carnegie Mellon University; Carnegie Mellon, 2012–

ANATOLE GERSHMAN, Distinguished Service Professor, Language Technologies Institute – Ph.D., Yale University; Carnegie Mellon, 2007–

HARTMUT GEYER, Associate Professor, Robotics Institute – Ph.D., Friedrich-Schiller University; Carnegie Mellon, 2010–

PHIL GIBBONS, Professor, Computer Science Department – Ph.D., University of California at Berkeley; Carnegie Mellon, 2015–
Carnegie Mellon, 2014–
EDUARD HOVY, Research Professor, Language Technologies Institute –
Ph.D., University Of California at Berkeley; Carnegie Mellon, 2004–
JASON HONG, Associate Professor, Human-Computer Interaction Institute –
University of Colorado; Carnegie Mellon, 1993–
RALPH HOLLIS, Research Professor Emeritus, Robotics Institute –
Ph.D., University of Colorado; Carnegie Mellon, 1993–
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–
MATTHEW JOHNSON-ROBERSON, Professor, Director, Robotics Institute –
Ph.D., University of Sydney; Carnegie Mellon, 2022–
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 - Ph.D, 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–
JAMES HERBSLEB, Director, Professor, Institute for Software Research –
Ph.D., Carnegie Mellon University; Carnegie Mellon, 2008–
CHRISTOPHER HARRINGTON, Assistant Professor, Human Computer Interaction Institute –
Ph.D., University Of Nebraska; Carnegie Mellon, 2010–
SEUNGJIN KIM, Assistant Professor, Computational Biology Department - Ph.D., University of California at Irvine; Carnegie Mellon, 2013–
KENNETH KOEDINGER, Professor, Human-Computer Interaction Institute –
Ph.D., Carnegie Mellon University; Carnegie Mellon, 2011–
SEUNGJIN KIM, Assistant Professor, Computational Biology Department - Ph.D.,
University of California; Carnegie Mellon, 1997–
CARL KINGSFORD, Professor, Computer Science Department – M.S.,
Princeton University; Carnegie Mellon, 2012–
KRIS KITANI, Associate Research Professor, Robotics Institute – Ph.D.,
University of Tokyo; Carnegie Mellon, 2016–
ANIKET KITTUR, Professor, Human-Computer Interaction Institute – Ph.D.,
University of California At Los Angeles; Carnegie Mellon, 2009–
DANIEL KLAG, Systems Scientist, Institute for Software Research - Ph.D.,
University of Basel; Carnegie Mellon, 2021–
JOSHUA KANGAS, Assistant Professor, Computational Biology Department –
Ph.D., University of Michigan; Carnegie Mellon, 2018–
OLIVER KROEMER, Assistant Professor, Robotics Institute – Ph.D.,
Technische Universität Darmstadt; Carnegie Mellon, 2017–
CHRINMAY KARNI, 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–
CL AIRE LE GOUES, Associate Professor, Institute for Software Research –
Ph.D., University of Virginia; Carnegie Mellon, 2013–
CHRISTIAN LEBIERE, Research Psychologist, Psychology – Ph.D.,
Carnegie Mellon University; Carnegie Mellon, 1999–
EUN SUN LEE, Associate Teaching Professor, Institute for Software Research –
Ph.D., 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–
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 WEHBE, 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–

BRYAN WILDER, Assistant Professor, Machine Learning Department - Ph.D., Harvard University; Carnegie Mellon, 2022–

DAVID WOODRUFF, Professor, Computer Science Department – Ph.D., Massachusetts Institute of Technology; Carnegie Mellon, 2017–

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

SHERRY TONGSHUAN WU, Assistant Professor, Human Computer Interaction Institute - Ph.D., Washington University, Saint Louis; Carnegie Mellon, 2022–

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–


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

WENZHEN YUAN, Assistant Professor, Robotics Institute - Ph.D., Massachusetts Institute of Technology; Carnegie Mellon, 2019–

HAIYI ZHU, Associate 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–