School of Computer Science

Tom Mitchell, Interim Dean
Guy Blelloch, Associate Dean for Undergraduate Education
Thomas Cortina, Assistant Dean for Undergraduate Education

Location: GHC 4115
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 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, the School of Computer Science 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 the School of Computer Science a world leader in research and education. Recently, the School of Computer Science launched two new undergraduate majors: Computational Biology and Artificial Intelligence. These new majors, along with the highly-ranked Computer Science major, give students in the School of Computer Science a distinct path 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
- 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://coursedepartment.web.cmu.edu/schools-colleges/schoolofcomputerscience/artificialintelligence) (B.S. degree)
- Computational Biology (http://coursedepartment.web.cmu.edu/schools-colleges/schoolofcomputerscience/undergraduatecomputationalbiology) (B.S. degree, additional major, minor)
- Computer Science (http://coursedepartment.web.cmu.edu/schools-colleges/schoolofcomputerscience/undergraduatecomputerscience) (B.S. degree, additional major, minor)
- SCS additional majors and minors (http://coursedepartment.web.cmu.edu/schools-colleges/schoolofcomputerscience/addimajorsminors)

Students who apply to, and are directly admitted into, the School of Computer Science can choose between three primary majors: Artificial Intelligence, Computational Biology or Computer Science. Students admitted into the School of Computer Science and the College of Fine Arts are also given the option to pursue a joint major in Computer Science and Art. Similarly prepared students from other Carnegie Mellon colleges are eligible to apply for internal transfer to the School of Computer Science (for Computational Biology or Computer Science) and will be considered for transfer if grades in specific requirements are sufficiently high and space is available. 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://coursedepartment.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 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 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 is occasionally continued on probation. 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 will 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 2.0.

The minimum period of suspension is one academic year (two non-summer semesters). 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 Assistant Dean for Undergraduate Education, 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 Assistant Dean for Undergraduate Education, 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.
Undergraduate students admitted to colleges at CMU other than SCS may not subsequently transfer into the Artificial Intelligence major for any changes to this policy at the start of each academic year.

Students may apply for transfer by the mid-semester break in the semester when the last of the six required courses will be completed. In the case of courses 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 a 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 office of the Assistant Dean for Undergraduate Education 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 academic performance, 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 a reflection essay. It is important to note that extremely few external transfers are admitted to the SCS program at Carnegie Mellon University. At this time, no transfers will be allowed into the Artificial Intelligence program for non-CMU students. External transfers who are admitted for Computer Science or Computational Biology may not subsequently transfer into the Artificial Intelligence program due to high demand within CMU.

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 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/schoolofcomputerscience/undergraduatecomputationalbiology) 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.

Artificial Intelligence and Computer Science majors
For Artificial Intelligence and Computer Science 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 can be taken by AI and CS majors to satisfy this requirement are given in the list below. (Consult your academic advisor for additional choices available each semester.)

02-223  Personalized Medicine: Understanding Your Own Genome (can be paired with a course in Biology 03-xxx for two courses in one department) 9
03-121  Modern Biology 9
03-125  Evolution 9
03-132  Basic Science to Modern Medicine 9
03-133  Neurobiology of Disease 9
06-100  Introduction to Chemical Engineering 12
06-221  Thermodynamics 9
09-105  Introduction to Modern Chemistry I 10
09-106  Modern Chemistry II 10
09-217  Organic Chemistry I 9
09-218  Organic Chemistry II 9
09-225  Climate Change: Chemistry, Physics and Planetary Science 9
12-100  Exploring CEE: Infrastructure and Environment in a Changing World 12
12-201  Geology 9
18-100  Introduction to Electrical and Computer Engineering 12
18-220  Electronic Devices and Analog Circuits 12
18-240  Structure and Design of Digital Systems 12
24-101  Fundamentals of Mechanical Engineering 12
24-231  Fluid Mechanics 10
24-261  Statics 10
24-351  Dynamics 10
33-114  Physics of Musical Sound 9
33-120  Science and Science Fiction 9
33-121  Physics I for Science Students 12
33-142  Physics II for Engineering and Physics Students 12
or 33-152  Matter and Interactions I 9
33-224  Stars, Galaxies and the Universe 9
42-101  Introduction to Biomedical Engineering 12
42-202  Physiology 9
42-341  Introduction to Biomechanics 9
85-219  Biological Foundations of Behavior (can be paired with a course in Biology 03-xxx for two courses in one department) 9

At present, courses meeting the lab requirement are:

02-261  Quantitative Cell and Molecular Biology Laboratory (can be paired with a course in Biology 03-xxx for two courses in one department) 9
03-124  Modern Biology Laboratory 9
09-101  Introduction to Experimental Chemistry (This 3 unit lab together with 09-105 satisfies the lab requirement.) 3
09-221  Laboratory I: Introduction to Chemical Analysis 12
27-100  Engineering the Materials of the Future 12
33-104  Experimental Physics 9
42-203  Biomedical Engineering Laboratory 9
85-310  Research Methods in Cognitive Psychology 9
85-314  Cognitive Neuroscience Research Methods 9

The following MCS and CIT courses cannot be used to satisfy the Science and Engineering requirement:

03-111  Computational Molecular Biology and Genomics 9
03-132  Computational Methods for Biological Modeling and Simulation 9
06-262  Mathematical Methods of Chemical Engineering 12
09-103  Atoms, Molecules and Chemical Change 9
09-231  Mathematical Methods for Chemists 9

12-271  Introduction to Computer Application in Civil & Environmental Engineering 9
18-090  Twisted Signals: Multimedia Processing for the Arts 10
18-200  ECE Sophomore Seminar 1
18-202  Mathematical Foundations of Electrical Engineering 12
18-213  Introduction to Computer Systems 12
18-345  Introduction to Telecommunication Networks 12
18-411  Computational Techniques in Engineering 12
18-482  Telecommunications Technology and Policy for the Internet Age 12
18-487  Introduction to Computer Security 12
18-540  Rapid Prototyping of Computer Systems 12
19-101  Introduction to Engineering and Public Policy 12
19-211  Ethics and Policy Issues in Computing 9
19-325  Technology and Policy Writing for Lay Audiences 9
19-402  Telecommunications Technology and Policy for the Internet Age 12
19-411  Global Competitiveness: Firms, Nations and Technological Change 9
19-432  Special Topics: Bitcoin and Cryptocurrencies 6
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

All Electrical and Computer Engineering graduate courses [18-6xx, 18-7xx, 18-8xx, 18-9xx] cannot be used 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. Consult with a CS 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/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) 9
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 Assistant Dean for Undergraduate Education, if the minis meet the goals of the desired category. **NOTE: Artificial Intelligence majors must replace Category 1 with Category 1A: Cognitive Studies which is a subset of Category 1.**

Category 1 (for Computational Biology and Computer Science majors): 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
80-130  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
Institutions organize individual preferences and actions into collective outcomes.

Category 1 (all SCS majors): Economic, Political and Social Institutions - this requirement explores the processes by which institutions organize individual preferences and actions into collective outcomes.

Category 1A (for Artificial Intelligence majors): Cognitive Studies - this requirement explores how the brain and the mind work.

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.
Complete 3 non-technical courses of at least 9 units each from any of the departments in the 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.

The most up-to-date list of additions and deletions can be found at http://www.cs.cmu.edu/content/bcs-humanities-and-arts-requirements and supersedes the lists given below. Consult with a CS undergraduate advisor for additional information.

Deletions

The following courses 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:

- 36-200 Reasoning with Data (9)
- 36-202 Statistics & Data Science Methods (9)
- 36-207 Probability and Statistics for Business Applications (9)
- 36-208 Regression Analysis (9)
- 36-217 Probability Theory and Random Processes (9)
- 36-220 Engineering Statistics and Quality Control (9)
- 36-225 Introduction to Probability Theory (9)
- 36-226 Introduction to Statistical Inference (9)
- 36-247 Statistics for Lab Sciences (9)
- 36-303 Sampling, Survey and Society (9)
- 36-304 Biostatistics (9)
- 36-309 Experimental Design for Behavioral & Social Sciences (9)
- 36-314 Biostatistics (9)
- 36-315 Statistical Graphics and Visualization (9)
- 36-326 Mathematical Statistics (Honors) (9)
- 36-350 Statistical Computing (9)
- 36-401 Modern Regression (9)
- 36-402 Advanced Methods for Data Analysis (9)
- 36-410 Introduction to Probability Modeling (9)
- 36-428 Time Series (6)
- 36-459 Statistical Models of the Brain (12)
- 36-461 Special Topics: Statistical Methods in Epidemiology (9)
- 36-462 Special Topics: Data Mining (9)
- 36-463 Special Topics: Multilevel and Hierarchical Models (9)
- 36-464 Special Topics: Applied Multivariate Methods (9)
- 36-468 Special Topics: Text Analysis (9)
- 36-490 Undergraduate Research (9)
- 36-492 Topic Detection and Document Clustering (6)
- 36-494 Astrostatistics (6)
- 51-224 CD: Web Design (9)
- 51-257 Introduction to Computing for Creative Practices (10)
- 51-327 Design Center: Introduction to Web Design (9)
- 51-328 Advanced Web Design (9)
- 67-240 Mobile Web Design & Development (9)
- 67-250 The Information Systems Milieux (9)
- 67-261 Information Design Fundamentals (9)
- 67-262 Database Design and Development (9)
- 67-272 Application Design and Development (9)
- 67-279 Introduction to Geographical Information Systems (6)
- 67-306 Special Topics: Management of Computer and Information Systems (6)
- 67-308 Innovation Studio: Health Care Information Systems (9)
- 67-309 Special Topics: Information Assurance and Security (6)
- 67-317 Mobile Web Development and Usability Testing (9)
- 67-319 Global Technology Consulting Groundwork (3)
- 67-324 Accelerating Innovation and Entrepreneurship (9)
- 67-327 Web Application Security (6)
- 67-328 Mobile to Cloud: Building Distributed Applications (9)
- 67-329 Contemporary Themes in Global Systems (9)
- 67-330 Technology Consulting in the Community (9)
- 67-331 Technology Consulting in the Global Community (3)
- 67-344 Organizational Intelligence in the Information Age (9)
- 67-353 IT & Environmental Sustainability (6)
- 67-364 Practical Data Science (9)
- 67-373 Information Systems Consulting Project (12)
- 67-390 Independent Study in Information Systems (Var.)
- 67-391 Independent Study in Information Systems (Var.)
- 67-440 IDeATe Mobile Application Design & Development (9)
- 67-442 Mobile Application Development in iOS (9)
- 67-475 Innovation in Information Systems (12)
- 67-490 Practicum in Information Systems (Var.)
- 73-230 Intermediate Microeconomics (9)
- 73-240 Intermediate Macroeconomics (9)
- 73-274 Econometrics I (9)
- 73-347 Game Theory for Economists (9)
- 73-374 Econometrics II (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)
- 80-222 Measurement and Methodology (9)
- 80-223 Causality and Probability (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)
- 80-405 Game Theory (9)
- 80-411 Proof Theory (9)
- 80-413 Category Theory (9)
- 80-521 Seminar on Formal Epistemology (Var.)
- 85-213 Human Information Processing and Artificial Intelligence (9)
- 85-219 Biological Foundations of Behavior (9)
- 85-370 Perception (9)
- 85-414 Cognitive Neuropsychology (9)
- 88-251 Empirical Research Methods (9)

Additions

The following courses outside of Dietrich College and the College of Fine Arts may count toward the Humanities and Arts requirement in SCS:

- 17-333 Privacy Policy, Law, and Technology (formerly 08-533) (9)
- 17-562 Law of Computer Technology (formerly 08-332) (9)
- 19-101 Introduction to Engineering and Public Policy (12)
- 19-402 Telecommunications Technology and Policy for the Internet Age (12)
- 19-411 Global Competitiveness: Firms, Nations and Technological Change (9)
- 32-102 Seapower and Maritime Affairs (6)
- 32-201 Leadership & Management (9)
- 32-402 Leadership and Ethics (9)
- 70-160 Graphic Media Management (9)
- 70-311 Organizational Behavior (9)
- 70-321 Negotiation and Conflict Resolution (9)
- 70-332 Business, Society and Ethics (9)
- 70-340 Business Communications (9)
- 70-341 Team Dynamics and Leadership (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 survey paper summarizing prior results 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 in an oral summary 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 assistant 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 research and innovation in computer science (15-300, 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 in computer science (15-400, 12 units) that allows students to complete a small-scale research study or experiment and present a research poster. students who use 15-400 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

students who need help finding potential advisors should get in touch with the associate dean or assistant dean for undergraduate education. applications to the program are due by the end of the semester prior to the start of the thesis, typically the end of the junior spring semester.

students successfully completing this thesis 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 adm0ni, 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–

j jonathan aldrich, professor, institute for software research – ph.d., university of washington; carnegie mellon, 2003–


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 bagnell, associate professor, robotics institute – ph.d., carnegie mellon university; carnegie mellon, 2004–

maria florina balcan, associate professor, machine learning department – ph.d., carnegie mellon university; carnegie mellon, 2014–

stephanie balzer, systems scientist, carnegie mellon university – ph.d., eth zurich; carnegie mellon, 2016–

ziv bar-jooseph, 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 bernstsen, associate teaching professor, human computer interaction institute – m.s., duquesne university; carnegie mellon, 2005–


alan black, professor, language technologies institute – ph.d., university of edinburgh; carnegie mellon, 1999–

guy blelloch, associate dean for undergraduate education and professor, computer science department – ph.d., massachusetts institute of technology; carnegie mellon, 1988–

l enore blum, distinguished career professor, computer science department – ph.d., massachusetts institute of technology; carnegie mellon, 1999–

manuel blum, university professor, computer science department – ph.d., massachusetts institute of technology; carnegie mellon, 1999–

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 p ennsylvania; 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–


RANDAL BRYANT, University Professor, Computer Science Department – Ph.D., Massachusetts Institute Of Technology; Carnegie Mellon, 1984–

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

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

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

JUSTINE CASSELL, Professor, Language Technologies Institute – Ph.D., of Technology; Carnegie Mellon, 2019–

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

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

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

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

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

JAIME CARBONELL, University Professor and Director, Language Technologies Institute – Ph.D., Yale University; Carnegie Mellon, 1979–

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

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 Torino; Carnegie Mellon, 2016–

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

NICOLAS CHRISTIN, Associate Professor - Ph.D., University of Virginia; Carnegie Mellon, 2017–

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

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

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

THOMAS CORTINA, Assistant Dean for Undergraduate Education and Teaching Professor, Computer Science Department – Ph.D., Politecnico University (NYU); Carnegie Mellon, 2004–

KEEAN CRANE, Assistant 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–

LAURA DABISH, Associate Professor, Human Computer Interaction Institute – Ph.D., Carnegie Mellon University; Carnegie Mellon, 2007–

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

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


ARTUR DUBRAWSKI, 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 ERDMANN, Professor, Robotics Institute – Ph.D., Massachusetts Institute Of Technology; Carnegie Mellon, 1989–

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

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, Department Head; Human-Computer Interaction Institute – Ph.D., Carnegie Mellon University; Carnegie Mellon, 2000–

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


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

CAROL FRIEZE, Director, Women@SCS and SCS4ALL, School of Computer Science - Ph.D., Carnegie Mellon University; Carnegie Mellon, 2000–

JOHN GALEOTTI, 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–

GARTH GIBSON, Professor, Computer Science Department – Ph.D., University Of California; Carnegie Mellon, 1991–

IOANNIS GKIOLEKAS, Assistant Professor, Robotics Institute – Ph.D., Harvard; Carnegie Mellon, 2017–

CLARK GLYMOUR, University Professor – Ph.D., Indiana University; Carnegie Mellon, 1985–

MAYANK GOEL, Assistant Professor, Institute for Software Research – Ph.D., University of Washington; Carnegie Mellon, 2016–

SETH GOLSTEIN, Associate Professor, Computer Science Department – Ph.D., University Of California; Carnegie Mellon, 1997–

GEOFFREY GORDON, Professor, Machine Learning Department – Ph.D., Carnegie Mellon University; Carnegie Mellon, 2001–

MATTHEW GORMLEY, Assistant Teaching Professor, Machine Learning Department – Ph.D., John Hopkins University; Carnegie Mellon, 2015–

VIPUL Goyal, Associate Professor, Computer Science Department – Ph.D., University of California at Los Angeles; Carnegie Mellon, 2017–

MATTHIAS GRABMAIR, Systems Scientist, Language Technologies Institute – Ph.D., University of Pittsburgh; Carnegie Mellon, 2015–

ASHINAV GUPTA, Associate Professor, Robotics Institute – Ph.D., University of Maryland; Carnegie Mellon, 2011–

ANUPAM GUPTA, Professor, Computer Science Department – Ph.D., University Of California at Berkeley; Carnegie Mellon, 2003–

VENKATESAN GURUSWAMI, Professor, Computer Science Department – Ph.D., Massachusetts Institute of Technology; Carnegie Mellon, 2009–

BERNARD HAEUPLER, Associate Professor, Computer Science Department – Ph.D., Massachusetts Institute of Technology; Carnegie Mellon, 2009–

JESSICA HAMMER, Assistant Professor, Human-Computer Interaction Institute – Ph.D., Columbia University; Carnegie Mellon, 2014–

MOR HARCHOL-BALTER, Professor, Computer Science Department – Ph.D., University Of California at Berkeley; Carnegie Mellon, 1999–

ROBERT HARPER, Professor, Computer Science Department – Ph.D., Cornell University; Carnegie Mellon, 1988–

ROBERT HARPER, Professor, Computer Science Department – Ph.D., Cornell University; Carnegie Mellon, 1988–

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

KATE FRAGKIADAKI, Assistant Professor, Machine Learning Department – Ph.D., University of Pennsylvania; Carnegie Mellon, 2016–
CHRISTOPHER HARRISON, Assistant Professor, Human-Computer Interaction Institute – Ph.D., Carnegie Mellon University; Carnegie Mellon, 2014–

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

MARTIAL HEBERT, Professor and Director, Robotics Institute – Ph.D., Paris-XI; Carnegie Mellon, 1984–

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

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

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

MICHAEL HILTON, Assistant 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, Assistant 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–

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

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

EUNSU 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, Senior Systems Scientist, 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, Assistant Professor, Human Computer Interaction Institute – Ph.D., Ohio State University; Carnegie Mellon, 2015–

DILSUN KAYNUR, Assistant 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–

SARA KIESLER, Professor Emeritus, Human Computer Interaction Institute – Ph.D., Ohio State University; Carnegie Mellon, 1979–

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, 2016–

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–

PRAVEESH 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, Assistant 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 LEBIERE, Research Psychologist, Psychology – Ph.D., Carnegie Mellon University; Carnegie Mellon, 1999–

EUN SUN LEE, Assistant 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–

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

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

JIAN MA, Associate 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–

MATTHEW MASON, Professor, Robotics Institute – 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, Computer Science Department – Ph.D., University Of California; Carnegie Mellon, 1988–

HEATHER MILLER, Assistant Professor, Institute for Software Research - Ph.D., Ecole 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–

ANDREW MOORE, Dean and Professor, School of Computer Science – Ph.D., University of Cambridge; Carnegie Mellon, 1993–

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