Donald Bren School of Information and Computer Sciences

On This Page:

Hal S. Stern, Dean
6210 Donald Bren Hall
Academic Counseling: 949-824-5156
http://www.ics.uci.edu/

Overview

The Donald Bren School of Information and Computer Sciences (ICS) embodies excellence, creativity, and collaborative innovation in computer science and information technology. As the only independent computing school in the University of California system, it is well-positioned to continue its tradition of exploring and advancing the boundaries of a broad, multidisciplinary field on a global scale.

ICS faculty have extensive training in traditional computer science, as well as engineering, mathematics and statistics, and the social sciences. The School's stand-alone structure, as opposed to being part of an engineering school, enables the faculty to take the broadest possible view of computer science and information technology. This breadth is reflected in the diverse set of academic degree options for undergraduate and graduate students, some of which are interdisciplinary and jointly administered with other academic units.

The School's three departments — Computer Science, Informatics, and Statistics — fuel a wide range of instructional and research efforts, including: design of algorithms and data structures; computer architecture and embedded computer systems; networked and distributed systems; systems software; social and mobile computing; artificial intelligence, machine learning and data mining; computer games and virtual worlds; databases and information retrieval; computer graphics and visualization; bioinformatics, computational biology and genomics; computer-supported cooperative work, human-centered computing and human-computer interaction; security and privacy; software engineering; managerial and social aspects of computing technology; and statistics. The vibrant Bren School community continues to explore innovative topics ranging from building complete computer systems on chips smaller than a human fingernail to developing user-interface systems that allow workers on opposite sides of the world to collaborate effectively. Bren School research continues to focus on how computing and information technology can be used to solve a broad set of real-world problems, such as improving how first responders communicate during a crisis, optimizing transportation systems, analyzing data to expedite biological research, and improving network security.

ICS faculty actively lead and participate in a variety of research laboratories, institutes and centers, including the Institute for Genomics and Bioinformatics; Institute for Software Research; Institute for Virtual Environments and Computer Games; California Institute for Telecommunications and Information Technology (Calit2); Data Science Initiative; Center for Machine Learning and Intelligent Systems; Center for Digital Transformation; Center for Emergency Response Technologies; Center for Algorithms and Theory of Computation; Center for Research in Sustainability, Collapse-Preparedness & Information Technology; Secure Computing and Networking Center; Center for Ethnography; Social & Technological Action Research Group; Secure Systems and Software Laboratory; Software Engineering and Analysis Lab; Computational Vision Lab; Transformative Play Lab; and Laboratory for Ubiquitous Computing and Interaction.

Faculty and student-driven research in the Bren School is supported through a variety of grants, gifts, and contracts from public and private institutions such as the State of California, the U.S. Department of Education, various U.S. defense agencies, the National Science Foundation, the National Institutes of Health, NASA, and various companies, including The Aerospace Corporation, Boeing, Disney, Experian, Google, IBM, Intel, Microsoft, Samsung, and Yahoo!

Faculty and alumni of the Bren School of ICS have contributed some of computing's most significant advancements, including revolutionizing computer-aided drafting techniques; the creation of the current Hypertext Transfer Protocol (HTTP/1.1); development of the Internet standards for HTTP and Uniform Resource Identifiers (URI); the founding of the Apache HTTP Server Project that produces the software for more than 60 percent of public Internet websites; and the creation of the Domain Name System (DNS) that translates Web and e-mail addresses into the numeric system used to route information along the Internet.

ICS is committed to increasing diversity in the computing and information technology fields. The Office of Access and Inclusion was created in 2014 as a joint initiative between ICS and UCI's Samueli School of Engineering to support the recruitment, retention, and graduation of undergraduate and graduate students from populations underrepresented in engineering and computer science. The School is also an active partner of the National Center for Women & Information Technology (NCWIT), whose overarching goal is parity in the professional information technology workforce, and a committed

BRAID (Building, Recruiting And Inclusion for Diversity) Institution, working to increase the percentage of women and students of color majoring in computer science.

Degrees

Business Information Management[1]	B.S.
Computer Game Science	B.S.
Computer Science	B.S., M.S., Ph.D.
Computer Science and Engineering[2]	B.S.
Data Science	B.S.
Human Computer Interaction and Design	M.H.C.I.D.
Informatics	B.S., Ph.D.
Information and Computer Science	B.S., M.S., Ph.D.[3]
Networked Systems[2]	M.S., Ph.D.
Software Engineering	B.S., M.S., Ph.D.
Statistics	M.S., Ph.D.

1 Offered jointly with The Paul Merage School of Business. See the Interdisciplinary Studies section of the *Catalogue* for information.

2 Offered jointly with The Henry Samueli School of Engineering. See the Interdisciplinary Studies section of the *Catalogue* for information.

3 Admission to the Ph.D. program is no longer available.

Honors

Honors at graduation, e.g., *cum laude, magna cum laude, summa cum laude*, are awarded to approximately the top 12 percent of the graduating seniors. A general criterion is that a student must have completed at least 72 units in residence at the University of California. The student's cumulative record at the end of the final quarter is the basis for consideration of awarding Latin Honors. Other important factors are considered visit at Honors Recognition.

Careers

Graduates of the Donald Bren School of Information and Computer Sciences go on to pursue a variety of careers in both industry and academia in the areas of cutting-edge technology, science and business. With the goal of solving real-world problems with a global impact, ICS graduates find limitless opportunities as leaders in virtually every domain—from aerospace, automotive, biomedical, business information management, consumer products, cybersecurity, data science, engineering, entertainment, environmental, finance, gaming, national defense, pharmacology, and software engineering. ICS graduates often find jobs as members of research and development teams—building advanced technologies; designing software and hardware systems; analyzing and securing data; and specifying, designing, and maintaining computing infrastructures for a variety of institutions—while others venture off to form successful start-up companies or work as independent consultants. It is also common for ICS graduates to spend a few years in their related industry before moving into management or advanced technical positions, while others find that the undergraduate educational experience in ICS is the perfect stepping-stone for pursuing graduate studies in various computer science, informatics and statistics degrees, or venturing off into other academic areas such as medicine, law, engineering, or management.

Undergraduate Programs

A Bren School of ICS undergraduate education is a blend of scholarship, science, technology, and practical application that forms an excellent foundation for professional life.

The basis of the undergraduate programs are a set of fundamental courses in mathematics and computer science, supplemented by general education courses from other academic disciplines. A premium is placed on both communication and quantitative skills. Students quickly gain hands-on experience with advanced computing systems, and intense use of computer and network technologies continues throughout the undergraduate program. Students study data organization, algorithm design and analysis, design and organization of hardware and network systems, software engineering, artificial intelligence, social aspects of system design and use, and management of technology. In the process, students work with state-of-the-art hardware and software technologies, and learn several contemporary programming languages.

The Bren School offers eight majors:

B.S. in Business Information Management (offered jointly with The Paul Merage School of Business)
B.S. in Computer Game Science
B.S. in Computer Science
B.S. in Computer Science and Engineering (offered jointly with The Henry Samueli School of Engineering)
B.S. in Data Science
B.S. in Informatics
B.S. in Information and Computer Science
B.S. in Software Engineering

The Bren School offers the following minors:

Bioinformatics
Digital Information Systems
Health Informatics
Informatics
Information and Computer Science
Statistics

Visit the ICS Student Affairs Office website for Majors and Minors restrictions (http://www.ics.uci.edu/ugrad/degrees/MajorMinor_Restrictions_Chart.pdf) and Double Major Restrictions (http://www.ics.uci.edu/ugrad/degrees/Dbl_Major_Restrictions_Chart.pdf).

Admissions

To ensure admission consideration for the fall quarter, students should be sure to file their application by November 30 of the prior year. The selection criteria include grades, test scores, and other considerations.

Transfer Student Policy

Transfer requirements vary by major.

Business Information Management
Computer Game Science
Computer Science
Computer Science and Engineering
Data Science
Informatics
Software Engineering

NOTE TO TRANSFER APPLICANTS: These majors require a series of lower-division courses, and prerequisites constrain the order in which they can be taken. Junior-level transfer students who must complete a significant part of this sequence may find that it will take longer than two years at UCI to complete their degree. Python, Java, and C++ are used in the curriculum; therefore, transfer students should plan to learn these languages by studying on their own or by completing related programming courses prior to their first quarter at UCI.

Change of Major

Students interested in changing their major to one offered by the School should contact the ICS Student Affairs Office for more information and assistance. Information is also available at the UCI Change of Major Criteria website (http://www.changeofmajor.uci.edu).

Major and Minor Restrictions

Bren School of ICS majors (including shared majors, BIM and CSE) pursuing minors within the Bren School of ICS may not count more than five courses toward both the major and minor. Some ICS majors and minors outside of the School are not permitted due to significant overlap. Visit the ICS Student Affairs Office website for Majors and Minors restrictions. (http://www.ics.uci.edu/ugrad/degrees/MajorMinor_Restrictions_Chart.pdf) All students should check the Double Major Restrictions Chart (http://www.ics.uci.edu/ugrad/degrees/Dbl_Major_Restr_Chart.pdf) and view our information page (http://www.ics.uci.edu/ugrad/degrees/Double_Majors.php) on double majoring to see what degree programs are eligible for double majoring.

Special Programs and Courses

The Bren School of ICS Honors Program

The Bren School of ICS Honors Program provides selected upper-division students an opportunity to carry out a research project under the direction of a faculty member in the School. Eligible students participate in the ICS Honors Seminar (I&C SCI H197), which provides an introduction to the range of current faculty research. Each student then affiliates with an ICS faculty advisor who agrees to supervise a minimum of two quarters of research. The participating student prepares a final written research report and submits a copy for review to both the faculty advisor and the Honors Program advisor. Successful completion of the Honors Program earns the student a certificate and medal from the School. Further, a notation of successful completion is added to the student's transcript. For more information about course requirements, application procedures, and deadlines visit http://honors.ics.uci.edu/, or contact the Student Affairs Office at 949 824-5156.

Other Opportunities

Bren School of ICS undergraduates may complement their educational experience by participating in other programs. Information about the following programs is available elsewhere in the *Catalogue* and via the program Web sites: Campuswide Honors Program, Undergraduate Research Opportunities Program, Education Road Program, and Student Achievement Guided by Experience (SAGE Scholars).

Concentration: Engineering and Computer Science in the Global Context

The globalization of the marketplace for information technology services and products makes it likely that ICS graduates will work in multicultural settings or be employed by companies with extensive international operations or customer bases. The goal of the concentration is to help students develop and

integrate knowledge of the history, language, and culture of a country or geographic region outside the United States, through course work both at UCI and an international host campus, followed by a technology-related internship in the host country.

All Bren School majors in good standing may propose an academic plan that demonstrates the ability to complete the concentration (a minimum of eight courses) and other requirements for graduation in a reasonable time frame. It is expected that a student's proposal will reflect a high degree of planning that includes the guidance of academic counselors and those at the UCI Study Abroad Center regarding course selection, as well as considerations related to internship opportunities, housing, and financial aid. Each student's proposed program of study must be approved by the Bren School of ICS Associate Dean for Student Affairs. The Associate Dean will be available to assist qualified students with the development of a satisfactory academic plan, as needed.

The concentration consists of the following components:

1. A minimum of eight courses at UCI or at the international campus with an emphasis on the culture, language (if applicable and necessary), history, literature of the country that corresponds to the international portion of the program, international law, international labor policy, global issues, global institutions, global conflict and negotiation, and global economics;
2. A one- or two-semester sequence of technical courses related to the major and, possibly, culture, history, and literature courses taken at an international university;
3. A two-month or longer technical internship experience in the same country as the international educational experience.

More information about the requirements for the concentration is available in the ICS Student Affairs Office.

Undergraduate Major in Business Information Management (BIM)

This program is administered jointly by the Bren School of ICS and The Paul Merage School of Business. For information, see the Interdisciplinary Studies section of the *Catalogue*.

Requirements for the B.S. Degree in Business Information Management

All students must meet the University Requirements. Major Requirements: See the Interdisciplinary Studies section of the *Catalogue.*

Undergraduate Major in Computer Game Science

The Computer Game Science major gives students a strong foundation in introductory information and computer science, an extensive education in technologies and design practices associated with computer games, and an opportunity to focus in two areas of particular interest to the student. Students who complete the major will be able to create interactive and human-centered game designs; implement games using skills in modeling, graphics, software engineering, hardware architectures, human interfaces, and aesthetics; and evaluate games and game technology for their use in education, art, and social change.

Career Paths. A wide variety of careers and graduate programs are open to Computer Game Science (CGS) graduates. The video game industry is comparable in size to the film and music industries, and job growth projections are strong for people with strong technical backgrounds. Many other fields, including mobile software development, interactive entertainment, and training and education software have demand for similar skill sets and knowledge. CGS graduates are well-trained in computer science, and can thus pursue graduate programs or any career that involves designing, implementing, evaluating, or interacting with computer-based systems.

Admissions

Freshman Applicants: See the Undergraduate Admissions section.

Transfer Applicants:

Junior-level applicants who satisfactorily complete course requirements will be given preference for admission. Applicants must satisfy the following requirements:

1. Complete one year of approved college-level math, preferably courses in calculus equivalent to UCI's MATH 2A - MATH 2B; if not available, two semester courses equivalent to other major-related math courses are acceptable.
2. Complete one year of transferable computer science courses involving concepts such as those found in Java, Python, C++, data structures, or other object-oriented or high-level programming language.

Transfer applicants to the Computer Game Science major should be aware that several lower-division courses must be taken at UCI; therefore, the minimum time to degree completion will exceed two years.

NOTE: The introductory sequence in ICS has moved to Python. The Bren School of ICS strongly encourages all participants to become familiar with this programming language prior to matriculation. Additional computer science courses beyond the two required are strongly recommended, particularly

those that align with the major(s) of interest. Java is used extensively in the curriculum; therefore, transfer students should plan to learn it by studying on their own or by completing a Java-related programming course prior to their first quarter at UCI.

Requirements for the B.S. Degree in Computer Game Science

All students must meet the University Requirements.

Major Requirements

Lower-division	
A. Select one of the two groups of courses:	
I&C SCI 21	Introduction to Computer Science I
I&C SCI 22	Introduction to Computer Science II
I&C SCI 45C	Programming in C/C++ as a Second Language
I&C SCI 46	Data Structure Implementation and Analysis
or	
I&C SCI 31	Introduction to Programming
I&C SCI 32	Programming with Software Libraries
I&C SCI 33	Intermediate Programming
I&C SCI 45C	Programming in C/C++ as a Second Language
I&C SCI 46	Data Structure Implementation and Analysis
B. Complete:	
I&C SCI 51	Introductory Computer Organization
C. Complete:	
IN4MATX 43	Introduction to Software Engineering
or I&C SCI 52	Introduction to Software Engineering
D. Complete:	
MATH 2A	Single-Variable Calculus
MATH 2B	Single-Variable Calculus
I&C SCI 6N	Computational Linear Algebra
or MATH 3A	Introduction to Linear Algebra
I&C SCI 6B	Boolean Algebra and Logic
I&C SCI 6D	Discrete Mathematics for Computer Science
STATS 67	Introduction to Probability and Statistics for Computer Science
E. Complete:	
I&C SCI 60	Computer Games and Society
I&C SCI 61	Game Systems and Design
I&C SCI 62	Game Technologies and Interactive Media
F. Complete:	
PHYSICS 3A	Basic Physics I
G. Complete:	
FLM&MDA 85A	Introduction to Film and Visual Analysis
or FLM&MDA 85C	New Media and Digital Technologies
Upper-division	
A. Computer Game Science Core Requirements	
I&C SCI 161	Game Engine Lab
I&C SCI 167	Multiplayer Game Systems
I&C SCI 168	Multiplayer Game Project
I&C SCI 169A- 169B	Capstone Game Project I and Capstone Game Project II
and select two of the following:	
I&C SCI 162	Modeling and World Building
I&C SCI 163	Mobile and Ubiquitous Games
I&C SCI 166	Game Design
B. Computer Science Core	
COMPSCI 112	Computer Graphics

COMPSCI 171	Introduction to Artificial Intelligence
C. Select two of the following:	
COMPSCI 122A	Introduction to Data Management
COMPSCI 132	Computer Networks
COMPSCI 143A	Principles of Operating Systems
COMPSCI 152	Computer Systems Architecture
IN4MATX 113	Requirements Analysis and Engineering
IN4MATX 121	Software Design: Applications
IN4MATX 131	Human Computer Interaction
D. CGS Elective Courses:	
Five additional courses chosen from those listed in E	
E. At least three of the 16 upper-division courses satisfying A–D must be in the same Bren ICS track.	

Bren ICS Tracks:

Algorithms	
COMPSCI 161	Design and Analysis of Algorithms
COMPSCI 162	Formal Languages and Automata
COMPSCI 163	Graph Algorithms
COMPSCI 164	Computational Geometry and Geometric Modeling
COMPSCI 165	Project In Algorithms And Data Structures
Artificial Intelligence	
COMPSCI 171	Introduction to Artificial Intelligence
COMPSCI 174	Bioinformatics
COMPSCI 175	Project in Artificial Intelligence
COMPSCI 177	Applications of Probability in Computer Science
COMPSCI 178	Machine Learning and Data-Mining
COMPSCI 179	Algorithms for Probabilistic and Deterministic Graphical Models
Computational Biology	
COMPSCI 183	Introduction to Computational Biology
COMPSCI 184A	Representations and Algorithms for Molecular Biology
COMPSCI 184B	Probabilistic Modeling of Biological Data
COMPSCI 184C	Computational Systems Biology
Computer Graphics and Vision	
COMPSCI 111	Digital Image Processing
COMPSCI 112	Computer Graphics
COMPSCI 114	Projects in Advanced 3D Computer Graphics
COMPSCI 116	Computational Photography and Vision
COMPSCI 117	Project in Computer Vision
Computer Networks	
COMPSCI 131	Parallel and Distributed Computing
COMPSCI 132	Computer Networks
COMPSCI 133	Advanced Computer Networks
COMPSCI 134	Computer and Network Security
COMPSCI 137/IN4MATX 124	Internet Applications Engineering
Databases	
COMPSCI 121/IN4MATX 141	Information Retrieval
COMPSCI 122A	Introduction to Data Management
COMPSCI 122B	Project in Databases and Web Applications
COMPSCI 125	Next Generation Search Systems
Hardware	
COMPSCI 145	Embedded Software
COMPSCI 151	Digital Logic Design

COMPSCI 153	Logic Design Laboratory
COMPSCI 154	Computer Design Laboratory
Human-Computer Interaction	
IN4MATX 131	Human Computer Interaction
IN4MATX 132	Project in Human-Computer Interaction Requirements and Evaluation
IN4MATX 133	User Interaction Software
IN4MATX 134	Project in User Interaction Software
Operating Systems	
COMPSCI 143A	Principles of Operating Systems
COMPSCI 143B	Project in Operating System Organization
COMPSCI 144	High-performance Computers and Program Optimization
COMPSCI 146	Programming in Multitasking Operating Systems
Programming Languages and Compilers	
IN4MATX 101	Concepts in Programming Languages I (same as COMPSCI 141)
IN4MATX 102	Concepts of Programming Language II
COMPSCI 142A	Compilers and Interpreters
COMPSCI 142B	Language Processor Construction
Simulation and Optimization	
COMPSCI 115	Computer Simulation
COMPSCI 169	Introduction to Optimization
Social and Organizational Computing	
IN4MATX 151	Project Management
IN4MATX 153	Computer Supported Cooperative Work
IN4MATX 161	Social Analysis of Computing
IN4MATX 162W	Organizational Information Systems
IN4MATX 163	Project in the Social and Organizational Impacts of Computing
IN4MATX 171	Introduction to Medical Informatics
Software Engineering	
IN4MATX 113	Requirements Analysis and Engineering
IN4MATX 115	Software Testing, Analysis, and Quality Assurance
IN4MATX 117	Project in Software System Design
IN4MATX 121	Software Design: Applications
IN4MATX 122	Software Design: Structure and Implementation
IN4MATX 124	Internet Applications Engineering (same as COMPSCI 137)
Non-Track Courses (some of these courses have prerequisites that are not part of the CGS major):	
Computer Game Science	
I&C SCI 162	Modeling and World Building
I&C SCI 163	Mobile and Ubiquitous Games
I&C SCI 166	Game Design
Business Management	
MGMT 101	Management Science
MGMT 102	Managing Organizational Behavior
MGMT 105	Introduction to Marketing
MGMT 154	Global Marketing
MGMT 155	Brand Management
Cognitive Science	
PSYCH 130A	Perception and Sensory Processes
PSYCH 131A	Vision
PSYCH 131B	Hearing
PSYCH 135M	The Mind/Body Problem
PSYCH 140C	Cognitive Science
Mathematics	

MATH 112A- 112B- 112C	Introduction to Partial Differential Equations and Applications and Introduction to Partial Differential Equations and Applications and Introduction to Partial Differential Equations and Applications
MATH 115	Mathematical Modeling
MATH 121A- 121B	Linear Algebra and Linear Algebra
Film and Media Studies	
FLM&MDA 113	Narrative/Image
FLM&MDA 114	Film, Media, and the Arts
FLM&MDA 117A	Introduction to Screenwriting
FLM&MDA 144	Studies in New Media

With prior approval of the ICS Associate Dean for Student Affairs, a student may design a new track, or an Independent Study, Honors Research, or Special Topics course may be substituted for a course in a track. Computer Game Science (CGS) elective courses may not be counted as part of the Management minor.

Major and minor restrictions: Click on the "Majors/Minors Restrictions" tab at the top of this page.

Sample Program of Study — Computer Game Science

Freshman		
Fall	**Winter**	**Spring**
I&C SCI 31	I&C SCI 32	I&C SCI 33
I&C SCI 60[1]	I&C SCI 61	I&C SCI 62
MATH 2A	MATH 2B	I&C SCI 6B
	WRITING 39B	WRITING 39C
Sophomore		
Fall	**Winter**	**Spring**
I&C SCI 45C	I&C SCI 46	IN4MATX 43
I&C SCI 51	I&C SCI 161	STATS 67
I&C SCI 6N	COMPSCI 112	General Education III/VII
	I&C SCI 6D	
Junior		
Fall	**Winter**	**Spring**
I&C SCI 162, 163, or 166[3]	I&C SCI 167	I&C SCI 162, 163, or 166[3]
COMPSCI 171	COMPSCI 122A, IN4MATX 113, IN4MATX 121, or IN4MATX 131[3]	I&C SCI 168
FLM&MDA 85A or 85C[2]	U-D Writing	Computer Game Science Elective
PHYSICS 3A		
Senior		
Fall	**Winter**	**Spring**
I&C SCI 169A	I&C SCI 169B	Computer Game Science Elective
General Education III	Computer Game Science Elective	General Education VI
General Education IV	General Education IV	General Education VIII

1 Fulfills GE III. Note, you must satisfy the UC Entry Level Writing requirement to enroll in I&C SCI 60.

2 Fulfills GE IV.

3 Select two of these.

NOTES:

1. Students are advised that this sample program lists the minimum requirements; it is possible that students may have to take additional courses to prepare for required courses.

2. The lower-division writing requirement must be completed by the end of the seventh quarter at UCI.

3. This is only a sample plan. Course offerings may be moved due to unforeseen circumstances. It is strongly recommended that students meet with an academic advisor to create an academic plan tailored to meet their specific areas of interest.

Undergraduate Major in Information and Computer Science

The Information and Computer Science major is intended for highly motivated students who are currently enrolled at UCI, who find that their academic and career interests are not well served by any of the existing ICS majors, and would be better served by a uniquely designed course of study.

Application Process

New students are not admitted directly to the Information and Computer Science major. Continuing students can apply to change their major to Information and Computer Science no earlier than the fall quarter of their sophomore year. Students must submit a proposal for a four-year plan of study, along with rationale explaining why the proposed plan is a well-motivated and coherent set of courses that does not fit into any of the existing ICS majors. Students submitting proposals are strongly encouraged to follow the lower-division requirements for one of the ICS majors (or provide a rationale for why this is not appropriate) and should include at least 48 units of upper-division ICS, Computer Science, Informatics, or Statistics courses. Proposals must be approved by the ICS Associate Dean for Student Affairs. See the ICS Student Affairs Office for more details. Complete information about changing majors to ICS is available at the UCI Change of Major Criteria website (http://www.changeofmajor.uci.edu).

Admissions

New students are not admitted directly to the Information and Computer Science major.

Transfer Applicants:

Students are strongly encouraged to follow the transfer preparation guidelines for any of the other Bren ICS majors.

Requirements for the B.S. Degree in Information and Computer Science

All students must meet the University Requirements. Major Requirements: See the ICS Student Affairs Office.

Undergraduate Major in Software Engineering

The Software Engineering major gives students a strong foundation in software engineering as well as a solid basis in computer science. Students who complete the major will be able to be productive members of software engineering teams in a variety of application domains including, but not restricted to, Web and mobile applications. The acquired technical knowledge and appreciation for life-long learning, combined with the ability to place software in the social context in which it is developed, empowers students to create novel applications that have the potential to bring social change.

Admissions

Freshman Applicants: See the Undergraduate Admissions section.

Transfer Applicants:

Junior-level applicants who satisfactorily complete course requirements will be given preference for admission. Applicants must satisfy the following requirements:

1. Completion of one year of approved college-level math, preferably courses in calculus equivalent to UCI's MATH 2A-MATH 2B; if not available, one year of coursework equivalent to other major-related math courses is acceptable.
2. Completion of one year of transferable Computer Science courses such as those found in Java, Python, C++, data structures, or other object-oriented or high-level programming language.

NOTE: The introductory sequence in ICS has moved to Python. The Bren School of ICS strongly encourages all participants to become familiar with this programming language prior to matriculation. Additional computer science courses beyond the two required are strongly recommended, particularly those that align with the major(s) of interest. Java is used extensively in the curriculum; therefore, transfer students should plan to learn it by studying on their own or by completing a Java-related programming course prior to their first quarter at UCI.

Additional courses beyond those required for admission must be taken to fulfill the lower-division degree requirements, as many are prerequisites for upper-division courses. For some transfer students, this may mean that it will take longer than two years to complete their degree.

Requirements for the B.S. Degree in Software Engineering

All students must meet the University Requirements.

Major Requirements

Lower-division

A. Select one of the following:	
I&C SCI 21- 22	Introduction to Computer Science I and Introduction to Computer Science II
or	
IN4MATX 41- 42	Informatics Core Course I and Informatics Core Course II
or	
I&C SCI 31- 32- 33	Introduction to Programming and Programming with Software Libraries and Intermediate Programming

B. Complete:	
I&C SCI 45C	Programming in C/C++ as a Second Language
C. Complete:	
I&C SCI 45J	Programming in Java as a Second Language
D. Complete:	
I&C SCI 46	Data Structure Implementation and Analysis
E. Complete:	
IN4MATX 43	Introduction to Software Engineering
or I&C SCI 52	Introduction to Software Engineering
F. Complete:	
I&C SCI 51	Introductory Computer Organization
G. Complete:	
MATH 2A- 2B	Single-Variable Calculus and Single-Variable Calculus
I&C SCI 6B	Boolean Algebra and Logic
I&C SCI 6D	Discrete Mathematics for Computer Science
I&C SCI 6N	Computational Linear Algebra
or MATH 3A	Introduction to Linear Algebra
STATS 67	Introduction to Probability and Statistics for Computer Science
Upper-division	
A. Core Requirements	
COMPSCI 122A	Introduction to Data Management
COMPSCI 143A	Principles of Operating Systems
COMPSCI 132	Computer Networks
COMPSCI 161	Design and Analysis of Algorithms
IN4MATX 101/COMPSCI 141	Concepts in Programming Languages I
IN4MATX 113	Requirements Analysis and Engineering
IN4MATX 115	Software Testing, Analysis, and Quality Assurance
IN4MATX 131	Human Computer Interaction
IN4MATX 121	Software Design: Applications
IN4MATX 122	Software Design: Structure and Implementation
IN4MATX 123	Software Architecture
or IN4MATX 124	Internet Applications Engineering
IN4MATX 151	Project Management
IN4MATX 191A	Senior Design Project
IN4MATX 191B	Senior Design Project
I&C SCI 139W	Critical Writing on Information Technology
B. Select four of the following:	
IN4MATX 102	Concepts of Programming Language II
IN4MATX 125/COMPSCI 113	Computer Game Development
IN4MATX 132	Project in Human-Computer Interaction Requirements and Evaluation
IN4MATX 133	User Interaction Software
IN4MATX 134	Project in User Interaction Software
IN4MATX 141/COMPSCI 121	Information Retrieval
IN4MATX 143	Information Visualization
IN4MATX 148	Project in Ubiquitous Computing
IN4MATX 161	Social Analysis of Computing
COMPSCI 133	Advanced Computer Networks
COMPSCI 134	Computer and Network Security
COMPSCI 142A	Compilers and Interpreters
COMPSCI 142B	Language Processor Construction

COMPSCI 145- 145L	Embedded Software and Embedded Software Laboratory
COMPSCI 146	Programming in Multitasking Operating Systems
COMPSCI 165	Project In Algorithms And Data Structures
I&C SCI 160	Graphics Processors and Game Platforms
I&C SCI 167	Multiplayer Game Systems
I&C SCI 168	Multiplayer Game Project

Software Engineering elective courses may not be counted as part of the Management minor.

Sample Program of Study — Software Engineering

Freshman		
Fall	**Winter**	**Spring**
I&C SCI 31	I&C SCI 32	I&C SCI 33
MATH 2A	MATH 2B	IN4MATX 43
WRITING 39A	WRITING 39B	I&C SCI 6B
	General Education III	WRITING 39C
Sophomore		
Fall	**Winter**	**Spring**
I&C SCI 45C	I&C SCI 46	COMPSCI 122A
I&C SCI 51	IN4MATX 113	COMPSCI 143A
I&C SCI 6D	I&C SCI 6N	STATS 67
General Education III/VII	IN4MATX 131	General Education III
Junior		
Fall	**Winter**	**Spring**
I&C SCI 45J	IN4MATX 122	IN4MATX 124
IN4MATX 115	IN4MATX 151	COMPSCI 132
IN4MATX 121	COMPSCI 161	General Education IV
General Education IV	General Education IV/VIII	
Senior		
Fall	**Winter**	**Spring**
IN4MATX 101	IN4MATX 191B	Software Engineering Elective
IN4MATX 191A	I&C SCI 139W	Software Engineering Elective
Software Engineering Elective	Software Engineering Elective	General Education VI
General Educaiton III		

NOTES:

1. Students are advised that this sample program lists the minimum requirements; it is possible that students may have to take additional courses to prepare for required courses.

2. The lower-division writing requirement must be completed by the end of the seventh quarter at UCI.

3. This is only a sample plan. Course offerings may be moved due to unforeseen circumstances. It is strongly recommended that students meet with an academic advisor to create an academic plan tailored to meet their specific areas of interest.

Important Notes:

Students enrolled in other degree programs who are interested in the field of computer science may pursue the Bren School introductory course sequences (I&C SCI 31/CSE 41, I&C SCI 32/CSE 42, and I&C SCI 33/CSE 43) followed by other courses for which they have met the prerequisites as far as their interests require and their programs permit. The introductory courses, along with other lower-division ICS courses, may be used to fulfill General Education requirements. Nonmajors may also take other Bren ICS courses for which they have met the prerequisites.

The ICS Student Affairs Office is staffed by professional academic counselors and peer advisors. These individuals are available to assist students with program planning, questions on University and School policies and procedures, progress toward graduation, and other issues that arise in the course of a student's education. **Faculty also are available for advising, generally for suggestions of additional course work in the student's academic, research, and career interest areas and on preparation for graduate school.**

Minor in Information and Computer Science

Students outside the School may also pursue a minor in Information and Computer Science. The minor provides a focused study of Information and Computer Science to supplement a student's major program of study and prepares students for a profession, career, or academic pursuit in which computer science is an integral part but is not the primary focus. The ICS minor contributes to students' competence in computing technology and

proficiency in programming as well as exposing them to the fundamentals of computer science. The minor allows students sufficient flexibility to pursue courses that complement their major field or address specific interests.

Requirements for the Minor in Information and Computer Science

Select one of the following groups:	
I&C SCI 21- 22- 46	Introduction to Computer Science I and Introduction to Computer Science II and Data Structure Implementation and Analysis
or	
I&C SCI 31- 32- 33- 45C- 46	Introduction to Programming and Programming with Software Libraries and Intermediate Programming and Programming in C/C++ as a Second Language and Data Structure Implementation and Analysis
Complete:	
I&C SCI 6D	Discrete Mathematics for Computer Science
Complete one of:	
I&C SCI 51	Introductory Computer Organization
or I&C SCI 52	Introduction to Software Engineering
or IN4MATX 43	Introduction to Software Engineering
Select two upper-division from the following: [1]	
CS 111-144	
CS 151–177	
IN4MATX 101–102	
IN4MATX 111–119	
IN4MATX 123	Software Architecture
IN4MATX 125	Computer Game Development
IN4MATX 131	Human Computer Interaction
IN4MATX 132–134	
IN4MATX 141	Information Retrieval
IN4MATX 148	Project in Ubiquitous Computing
IN4MATX 153	Computer Supported Cooperative Work
IN4MATX 161–163	
IN4MATX 171	Introduction to Medical Informatics

[1] COMPSCI 190–199 and IN4MATX 190–199 may not be applied to the minor.

NOTES:

1. Visit the ICS Student Affairs Office website for Majors and Minors restrictions (http://www.ics.uci.edu/ugrad/degrees/MajorMinor_Restrictions_Chart.pdf).
2. A maximum of two courses may be taken for the Pass/Not Pass grade option toward any minor
3. There are no applications for a Bren ICS minor. Students must have a peer advisor or academic counselor of their major add the minor to their Degree Audit once they begin pursuing the minor.
4. To ensure you are certified for the minor at graduation, the minor must be on your Degree Audit and Application for Graduation. Your Student Affairs Office will certify your minor at time of graduation, so it is important to keep them updated on your academic progress.

On This Page:

Graduate Programs in Information and Computer Sciences

The Bren School of ICS offers M.S. and Ph.D. degrees in Computer Science, Informatics, Information and Computer Science, Networked Systems, Software Engineering, and Statistics.

ICS M.S. students must complete one of the following concentrations: Embedded Systems or Informatics (INF).

For additional information about the following graduate programs and requirements, click on these links: Computer Science; Informatics; Statistics; Software Engineering; Networked Systems, which is supervised by an interdepartmental faculty group from the Department of Computer Science in the Bren School and the Department of Electrical Engineering and Computer Science in The Henry Samueli School of Engineering. Information is available on the Interdisciplinary Studies section of the *Catalogue*.

Admission

Applicants will be evaluated on the basis of their prior academic record. Applicants for the M.S. degree are expected to have a bachelor's degree in computer science or a related field. Those who do not have an undergraduate degree in computer science may take the Computer Science Subject GRE test to demonstrate sufficient background in the field. Scores are reviewed on a case-by-case basis. Ph.D. applicants will additionally be evaluated in their potential for creative research and teaching in Information and Computer Sciences.

Applicants are expected to have (1) skills in computer programming at least equivalent to those obtained in college-level courses in programming and language development; (2) skills in mathematics equivalent to those obtained in complete college-level courses in logic and set theory, analysis, linear algebra and modern algebra, or probability and statistics; (3) data structures, analysis of algorithms, automata theory, or formal languages; and (4) computer architectures.

All applicants are evaluated on the materials submitted: letters of recommendation, official GRE test scores, official college transcripts, and personal statement. For more information, contact the ICS graduate counselor at 949-824-5156 or send email to gcounsel@ics.uci.edu.

Financial Assistance

Financial assistance is available to Ph.D. students in the form of fellowships, teaching assistantships, and research assistantships. Although assistance varies, it is the School's goal to support all entering Ph.D. students, subject to availability of funds. International students who are not citizens of countries where English is either the primary or dominant language, as approved by Graduate Council, and who apply for teaching assistantships must take one of the approved English proficiency examinations. More information is available in the Graduate Division section of the *Catalogue.*

Students with a Previously Earned Master's Degree

Credit for one or all required courses may be given at the time of admission to those students who have completed a master's degree in computer science or a closely related field. Course equivalency will be determined by the Bren School Associate Dean for Student Affairs following a written recommendation from a sponsoring research advisor. Research advisors can require that a student take additional courses when this is appropriate.

An additional M.S. degree will not be awarded if the student currently holds an M.S. degree in computer science or a related field from another university.

Course Substitutions

A student who has taken relevant graduate courses at UCI or another university may petition to have a specific course certified as equivalent to one which satisfies Bren School of ICS requirements. The petition should describe the course and should be approved by either the student's advisor or the instructor teaching the class, and by the Associate Dean for Student Affairs. Only two courses can be substituted.

Master of Science Program

The Bren School offers M.S. degrees in Computer Science, Information and Computer Science, Networked Systems, Software Engineering, and Statistics.

Students pursuing the M.S. in Information and Computer Science must complete a concentration in Embedded Systems or Informatics (INF).

For additional information about the following graduate programs and requirements, click on these links: Computer Science; Software Engineering; Statistics; Networked Systems.

M.S. students may select one of two options, the thesis plan or the comprehensive examination plan, as described below. The normative time for completion of the M.S. degree is two years. All study must be completed within four calendar years from the date of admission.

Plan I: Thesis Plan. The thesis option is available for graduate students who may wish to continue on to a Ph.D. program or those who wish to concentrate on a specific problem. To qualify for this option, students must be in good academic standing with their Department. The student must enroll in at least two quarters of Thesis Supervision (COMPSCI 298 or IN4MATX 298) that will substitute for two required courses as specified under the concentration area or specialization of choice. All required courses must be completed with a grade of B or better, and the student must write a research or thesis project. A committee of three faculty members (voting members of the Academic Senate) will guide the student and give final approval of the thesis. The committee will consist of an advisor (faculty member from the student's department) who is willing to supervise the thesis project, and

two other faculty members (one of which must be from the student's department) who are willing to serve on the committee as readers of the thesis. An oral presentation of the thesis to the committee will be required. Seminar courses that have an "S" suffix (e.g., 209S) do not count toward degree requirements.

Plan II: Comprehensive Examination Plan. The student completes the required units as specified under the concentration area. Each course must be completed with a grade of B or better. Seminar courses that have an "S" suffix (e.g., 209S) do not count toward degree requirements. The student must take a comprehensive examination given by ICS faculty. The examination covers the core requirements.

ICS Concentration in Embedded Systems—M.S.

The goal of this program is to prepare students for challenges in developing future embedded systems. These future systems will further integrate communications, multimedia, and advanced processors with complex embedded and real-time software for automotive, medical, telecommunications, and many other application domains. Furthermore, embedded systems are becoming parallel, deploying multiprocessor systems-on-a-chip and parallel application software. An in-depth knowledge of the underlying scientific and engineering principles is required to understand these advances and to contribute productively to development of such systems. This program helps students master embedded system fundamentals, advanced computer architecture and compilers, networking, security, embedded, parallel and distributed software, and computer graphics in a sequence of courses and labs. Students also complete a large embedded systems project and may choose to write a Master's thesis.

Required Courses

The following courses must be completed with a grade of B or better.

Select six of the following:

List A	
COMPSCI 244	Introduction to Embedded and Ubiquitous Systems
COMPSCI 250A	Computer Systems Architecture
COMPSCI 232	Computer and Communication Networks
COMPSCI 203	Network and Distributed Systems Security
COMPSCI 242	Parallel Computing
COMPSCI 250B	Modern Microprocessors
COMPSCI 230	Distributed Computer Systems
COMPSCI 243	High-Performance Architectures and Their Compilers

Select six additional courses in one of the following two ways:

1. For students pursuing the M.S. thesis option, two four-unit courses in Thesis Supervision (COMPSCI 298) plus four graduate courses taken from List A or the following List B
2. For all other students, six graduate courses taken from List A or the following List B

List B	
COMPSCI 241	Advanced Compiler Construction
COMPSCI 245	Software for Embedded Systems
COMPSCI 252	Introduction to Computer Design
EECS 211	Advanced System Software
COMPSCI 211A	Visual Computing
COMPSCI 248A/IN4MATX 241	Introduction to Ubiquitous Computing
IN4MATX 211	Software Engineering
IN4MATX 235	Advanced User Interface Architecture
COMPSCI 236	Wireless and Mobile Networking
COMPSCI 265	Graph Algorithms
EECS 223	Real-Time Computer Systems

M.S. Students who do not have an undergraduate degree in Computer Science or equivalent must also take COMPSCI 260.

Comprehensive Examination or Thesis

Each student must either (1) pass a comprehensive examination administered by the Embedded Systems faculty; or (2) submit a thesis for approval by a three-person committee consisting of an advisor (who is an ICS Embedded Systems full-time faculty member) and two other full-time faculty members (one of which must be from ICS).

ICS Concentration in Informatics (INF)—M.S.

Informatics is the interdisciplinary study of the design, application, use, and impact of information technology. It goes beyond technical design to focus on the relationship between information system design and use in real-world settings. These investigations lead to new forms of system architecture,

new approaches to system design and development, new means of information system implementation and deployment, and new models of interaction between technology and social, cultural, and organizational settings.

In the Donald Bren School of Information and Computer Sciences, Informatics is concerned with software architecture, software development, design and analysis, programming languages, ubiquitous computing, information retrieval and management, human-computer interaction, computer-supported cooperative work, and other topics that lie at the relationship between information technology design and use in social and organizational settings. Effective design requires an ability to analyze things from many different perspectives, including computer science, information science, organizational science, social science, and cognitive science. Relevant courses in those disciplines are therefore an integral part of the program and give this concentration a unique interdisciplinary flavor—which is imperative as the computing and information technology fields play such a pervasive role in our daily lives.

This degree program requires 48 units of coursework, including 24 units of core requirements, and 24 units of electives (of which up to 12 units may be used as independent study).

A. Complete the following required core courses:	
IN4MATX 261	Social Analysis of Computing
IN4MATX 231	User Interface Design and Evaluation
or IN4MATX 232	Research in Human-Centered Computing
IN4MATX 209S	Seminar in Informatics (twice, usually in first year)
B. Complete the following Research Methods core courses:	
IN4MATX 201	Research Methodology for Informatics
IN4MATX 203	Qualitative Research Methods in Information Systems
IN4MATX 205	Quantitative Research Methods in Information Systems
C. Select six electives in Informatics: [1]	

[1] A set of six elective courses at the graduate level. The selection of courses should form a coherent educational plan to be approved by the student's faculty advisor or by the program director in the case that the student is not working with a faculty advisor. Although the courses may be chosen from any graduate level courses on campus, it is recommended that at least three be chosen from within the school of ICS. Students may use up to 12 units of independent study (IN4MATX 298/IN4MATX 299) as electives.

Faculty

Shannon L. Alfaro, M.S. University of California, Irvine, *Lecturer of Computer Science*

Animashree Anandkumar, Ph.D. Cornell University, *Assistant Professor of Electrical Engineering and Computer Science; Computer Science* (statistical inference and learning of graphical models, scalable network algorithms)

Nader Bagherzadeh, Ph.D. University of Texas at Austin, *Professor of Electrical Engineering and Computer Science; Computer Science* (parallel processing, computer architecture, computer graphics, VLSI design)

Brigitte Baldi, Ph.D. Massachusetts Institute of Technology, *Lecturer of Statistics*

Pierre F. Baldi, Ph.D. California Institute of Technology, *UCI Chancellor's Professor of Computer Science; Biological Chemistry; Biomedical Engineering; Developmental and Cell Biology* (bioinformatics, computational biology)

Scott Bartell, Ph.D. University of California, Davis, *Associate Professor of Program in Public Health; Environmental Health Sciences; Social Ecology; Statistics*

Lubomir Bic, Ph.D. University of California, Irvine, *Professor of Computer Science; Electrical Engineering and Computer Science* (parallel and distributed computing, mobile agents)

Rebecca W. Black, Ph.D. University of Wisconsin-Madison, *Associate Professor of Informatics*

Geoffrey C. Bowker, Ph.D. University of Melbourne, *Professor of Informatics; Anthropology; Visual Studies* (values in design, social studies of databases, science and technology studies)

Elaheh Bozorgzadeh, Ph.D. University of California, Los Angeles, *Associate Professor of Computer Science; Electrical Engineering and Computer Science* (design automation and synthesis for embedded systems, VLSI CAD, reconfigurable computing)

Carter Butts, Ph.D. Carnegie Mellon University, *Professor of Sociology; Electrical Engineering and Computer Science; Statistics* (mathematical sociology, social networks, quantitative methodology, human judgment and decision making, economic sociology)

Michael Carey, Ph.D. University of California, Berkeley, *Donald Bren Professor of Information & Computer Sciences and Professor of Computer Science*

Yunan Chen, Ph.D. Drexel University, *Associate Professor of Informatics; Program in Public Health* (medical informatics, human-computer interaction)

Pai H. Chou, Ph.D. University of Washington, *Professor of Electrical Engineering and Computer Science; Computer Science* (embedded systems, wireless sensor systems, medical devices, real-time systems, hardware/software co-synthesis)

John L. Crawford, Media Artist and Software Designer, *Graduate Advisor and Associate Professor of Dance; Informatics* (dance film, interactive media, telematic performance, motion capture, digital arts)

Rina Dechter, Ph.D. University of California, Los Angeles, *Professor of Computer Science*

Brian C. Demsky, Ph.D. Massachusetts Institute of Technology, *Associate Professor of Electrical Engineering and Computer Science; Computer Science* (compiler programming, language software engineering, fault tolerance)

Michael B. Dillencourt, Ph.D. University of Maryland, College Park, *Professor of Computer Science*

John Christopher Dobrian, Ph.D. University of California, San Diego, *Professor of Music; Informatics*

Rainer B. Doemer, Ph.D. Dortmund University, *Associate Professor of Electrical Engineering and Computer Science; Computer Science* (system-level design, embedded computer systems, design methodologies, specification and modeling languages)

James P. Dourish, Ph.D. University College London, *Professor of Informatics; Computer Science* (human-computer interaction, computer-supported cooperative work)

Nikil D. Dutt, Ph.D. University of Illinois at Urbana–Champaign, *UCI Chancellor's Professor of Computer Science; Cognitive Sciences; Electrical Engineering and Computer Science* (embedded systems, computer architecture, electronic design automation, software systems, brain-inspired architectures and computing)

Magda S. El Zarki, Ph.D. Columbia University, *Professor of Computer Science; Electrical Engineering and Computer Science; Informatics* (telecommunications, networks, wireless communication, video transmission)

David A. Eppstein, Ph.D. Columbia University, *UCI Chancellor's Professor of Computer Science*

Julian Feldman, Ph.D. Carnegie Institute of Technology, *Professor Emeritus of Computer Science*

Charless C. Fowlkes, Ph.D. University of California, Berkeley, *Associate Professor of Computer Science; Cognitive Sciences; Electrical Engineering and Computer Science* (computer vision, machine learning, computational biology)

Michael S. Franz, Ph.D. Swiss Federal Institute of Technology in Zurich, *Professor of Computer Science; Electrical Engineering and Computer Science* (systems software, particularly compilers and virtual machines, trustworthy computing, software engineering)

Daniel H. Frost, M.S. University of California, Irvine, *Senior Lecturer of Computer Science; Informatics* (artificial intelligence, software engineering, computer graphics, teaching of programming)

Jean-Luc Gaudiot, Ph.D. University of California, Los Angeles, *Professor of Electrical Engineering and Computer Science; Computer Science* (parallel processing, computer architecture, processor architecture)

Daniel L. Gillen, Ph.D. University of Washington, *Professor of Statistics; Program in Public Health*

Tony D. Givargis, Ph.D. University of California, Riverside, *Professor of Computer Science* (embedded systems, platform-based system-on-a-chip design, low-power electronics)

Michael T. Goodrich, Ph.D. Purdue University, *UCI Chancellor's Professor of Computer Science; Electrical Engineering and Computer Science* (computer security, algorithm design, data structures, Internet algorithmics, geometric computing, graphic drawing)

Richard H. Granger, Ph.D. Yale University, *Professor Emeritus of Computer Science*

Judith Gregory, Ph.D. University of California, San Diego, *Associate Adjunct Professor of Informatics* (values in design, translational biomedical informatics, participatory design, design and emotion)

Vijay Gurbaxani, Ph.D. University of Rochester, *Taco Bell Chair in Information Technology Management and Professor of Paul Merage School of Business; Informatics* (economics of information systems management, impact of information technology on organization and market structure)

Stacey A. Hancock, Ph.D. Colorado State University, *Lecturer with Potential Security of Employment of Statistics*

Ian G. Harris, Ph.D. University of California, San Diego, *Associate Professor of Computer Science; Electrical Engineering and Computer Science* (hardware/software covalidation, manufacturing test)

Gillian Hayes, Ph.D. Georgia Institute of Technology, *Professor of Informatics; Education* (interactive and collaborative technology, human-computer interaction, computer-supported cooperative work, educational technology, ubiquitous computing)

Wayne B. Hayes, Ph.D. University of Toronto, *Associate Professor of Computer Science*

Dan S. Hirschberg, Ph.D. Princeton University, *Professor of Computer Science; Electrical Engineering and Computer Science* (analyses of algorithms, concrete complexity, data structures, models of computation)

Alexander T. Ihler, Ph.D. Massachusetts Institute of Technology, *Associate Professor of Computer Science*

Sandra S. Irani, Ph.D. University of California, Berkeley, *Professor of Computer Science*

Mizuko Ito, Ph.D. Stanford University, *John D. and Catherine T. MacArthur Foundation Chair in Digital Media and Learning and Professor in Residence of Anthropology; Education; Informatics* (ethnography, game studies, youth culture, learning sciences, online communities)

Jesse C. Jackson, M.A. University of Toronto, *Director of the Minor in Digital Arts and Assistant Professor of Art; Informatics*

Ramesh Chandra Jain, Ph.D. Indian Institute of Technology Kharagpur, *Donald Bren Professor of Information & Computer Sciences and Professor of Computer Science*

Stanislaw M. Jarecki, Ph.D. Massachusetts Institute of Technology, *Professor of Computer Science*

Ivan G. Jeliazkov, Ph.D. Washington University, *Associate Professor of Economics; Statistics*

Wesley O. Johnson, Ph.D. University of Minnesota, *Professor of Statistics*

James Jones, Ph.D. Georgia Institute of Technology, *Associate Professor of Informatics* (software engineering, software testing and analysis, debugging and fault localization, static and dynamic analysis, software visualization)

Scott A. Jordan, Ph.D. University of California, Berkeley, *Professor of Computer Science; Electrical Engineering and Computer Science* (pricing and differentiated services in the Internet, resource allocation in wireless networks, telecommunications policy)

David G. Kay, J.D. Loyola Marymount University, *Senior Lecturer of Informatics; Computer Science* (computer law, computer science education)

Dennis F. Kibler, Ph.D. University of California, Irvine, *Professor Emeritus of Computer Science*

Raymond O. Klefstad, Ph.D. University of California, Irvine, *Lecturer of Computer Science*

Cory P. Knobel, Ph.D. University of Michigan, *Assistant Adjunct Professor of Informatics* (interactive and collaborative technology, values in design, modes of knowledge representation, philosophy of science and technology)

Alfred Kobsa, Ph.D. University of Vienna, *Professor of Informatics; Computer Science* (user modeling, human-computer interaction, artificial intelligence, cognitive science, interdisciplinary computer science)

Peter O. Krapp, Ph.D. University of California, Santa Barbara, *Professor of Film and Media Studies; English; Informatics; Visual Studies* (digital culture, media history, cultural memory)

Jeffrey L. Krichmar, Ph.D. George Mason University, *Professor of Cognitive Sciences; Computer Science* (computational neuroscience, robotics)

Fadi J. Kurdahi, Ph.D. University of Southern California, *Director, Center for Embedded Computer Systems and Professor of Electrical Engineering and Computer Science; Computer Science* (VLSI system design, design automation of digital systems)

Richard H. Lathrop, Ph.D. Massachusetts Institute of Technology, *Professor of Computer Science* (modeling structure and function, machine learning, intelligent systems and molecular biology, protein structure/function prediction)

Marco Levorato, Ph.D. University of Padua, *Assistant Professor of Computer Science; Electrical Engineering and Computer Science*

Chen Li, Ph.D. Stanford University, *Professor of Computer Science*

Kwei-Jay Lin, Ph.D. University of Maryland, College Park, *Professor of Electrical Engineering and Computer Science; Computer Science* (real-time systems, distributed systems, service-oriented computing)

Cristina V. Lopes, Ph.D. Northeastern University, *Professor of Informatics; Computer Science* (programming languages, acoustic communications, operating systems, software engineering)

George S. Lueker, Ph.D. Princeton University, *Professor Emeritus of Computer Science*

Aditi Majumder, Ph.D. University of North Carolina at Chapel Hill, *Professor of Computer Science; Electrical Engineering and Computer Science* (novel displays and cameras for computer graphics and visualization, human-computer interaction, applied computer vision)

Gloria J. Mark, Ph.D. Columbia University, *Professor of Informatics* (computer-supported cooperative work, human-computer interaction)

Athina Markopoulou, Ph.D. Stanford University, *Associate Professor of Electrical Engineering and Computer Science; Computer Science* (networking—reliability and security, multimedia networking, measurement and control, design and analysis of network protocols and algorithms, internet reliability and security, multimedia streaming, network measurements and control)

Melissa Mazmanian, Ph.D. Massachusetts Institute of Technology, *Associate Professor of Informatics* (computer-mediated communication, organization studies, information and communication technologies in practice, social response to emerging technologies, work/non-work negotiations in the information age)

Gopi Meenakshisumdaram, Ph.D. University of North Carolina at Chapel Hill, *Professor of Computer Science; Electrical Engineering and Computer Science* (geometry and topology for computer graphics, image-based rendering, object representation, surface reconstruction, collision detection, virtual reality, telepresence)

Sharad Mehrotra, Ph.D. University of Texas at Austin, *Professor of Computer Science*

Eric D. Mjolsness, Ph.D. California Institute of Technology, *Professor of Computer Science; Mathematics* (applied mathematics, mathematical biology, modeling languages)

Bonnie A. Nardi, Ph.D. University of California, Irvine, *Professor of Informatics* (computer-supported collaborative work, human-computer interaction, computer-mediated communication, user studies methods, activity theory, cultural responses to technology development)

Emily Navarro, Ph.D. University of California, Irvine, *Lecturer of Informatics*

Alexandru Nicolau, Ph.D. Yale University, *Department Chair and Professor of Computer Science; Electrical Engineering and Computer Science* (architecture, parallel computation, programming languages and compilers)

Gary Olson, Ph.D. Stanford University, *Donald Bren Professor of Information & Computer Sciences and Professor of Informatics* (interactive and collaborative technology, human-computer interaction, computer-supported cooperative work)

Judith Olson, Ph.D. University of Michigan, *Donald Bren Professor of Information & Computer Sciences and Professor of Informatics; Paul Merage School of Business; Planning, Policy, and Design* (interactive and collaborative technology, human-computer interaction, computer-supported cooperative work)

Hernando C. Ombao, Ph.D. University of Michigan, *Professor of Statistics*

Donald J. Patterson, Ph.D. University of Washington, *Associate Professor of Informatics; Computer Science* (ubiquitous computing, pervasive computing, human-computer interaction, artificial intelligence, intelligent context for situated computing)

Richard Pattis, M.S. Stanford University, *Senior Lecturer of Computer Science; Informatics* (MicroWorlds for teaching programming, debugging, computational tools for non-computer scientists)

Simon G. Penny, M.F.A. Hong Kong University of Science and Technology, *Professor of Art; Informatics* (informatics, robotic sculpture, interactive environments, electronic media)

Kavita S. Philip, Ph.D. Cornell University, *Associate Professor of History; Comparative Literature; Informatics* (history of modern South Asia, science and technology, political ecology, critical theoretical studies of race, gender, colonialism, new media, and globalization)

Dale J. Poirier, Ph.D. University of Wisconsin-Madison, *Professor of Economics; Statistics*

David F. Redmiles, Ph.D. University of Colorado Boulder, *Professor of Informatics* (computer-supported cooperative work, human computer interaction, software engineering, globally distributed development teams, user interfaces, software tools)

Amelia C. Regan, Ph.D. University of Texas at Austin, *Professor of Computer Science*

Stephanie Reich, Ph.D. Vanderbilt University, *Associate Professor of Education; Informatics; Psychology and Social Behavior* (child development, parenting, peer interactions, media, program evaluation)

Debra J. Richardson, Ph.D. University of Massachusetts, *Professor of Informatics* (software engineering, program testing, life-cycle validation, software environments)

Ardalan Amiri Sani, Ph.D. Rice University, *Assistant Professor of Computer Science* (involves building efficient, high performance, and reliable systems)

Isaac D. Scherson, Ph.D. Weizmann Institute of Science, *Professor of Computer Science; Electrical Engineering and Computer Science* (parallel computing architectures, massively parallel systems, parallel algorithms, interconnection networks, performance evaluation)

Babak Shahbaba, Ph.D. University of Toronto, *Associate Professor of Statistics; Computer Science*

Phillip C-Y Sheu, Ph.D. University of California, Berkeley, *Professor of Electrical Engineering and Computer Science; Biomedical Engineering; Computer Science* (database systems, interactive multimedia systems)

Alice Silverberg, Ph.D. Princeton University, *Professor of Mathematics; Computer Science* (algebra and number theory)

Patrick J. Smyth, Ph.D. California Institute of Technology, *Professor of Computer Science; Statistics*

Thomas A. Standish, Ph.D. Carnegie Institute of Technology, *Professor Emeritus of Information and Computer Sciences* (software testing and analysis, software semantics and epistemology, programming and cognition, software comprehension)

Hal S. Stern, Ph.D. Stanford University, *Dean of the Donald Bren School of Information and Computer Sciences, Ted and Janice Smith Family Foundation Endowed Chair in Information and Computer Science, and Professor of Statistics; Cognitive Sciences*

Mark Steyvers, Ph.D. Indiana University, *Professor of Cognitive Sciences; Computer Science; Psychology and Social Behavior* (higher-order cognition, cognitive neuroscience, computational modeling, collective intelligence)

Joshua Tanenbaum, M.A. Simon Fraser University, *Acting Assistant Professor of Informatics* (digital games and narrative, tangible and wearable interaction, maker and DIY culture, nonverbal communication and virtual worlds)

Richard N. Taylor, Ph.D. University of Colorado Boulder, *Professor Emeritus of Informatics* (software engineering, user interfaces, environments, team support)

Alexander W. Thornton, B.S. University of California, Irvine, *Lecturer of Computer Science*

William M. Tomlinson, Ph.D. Massachusetts Institute of Technology, *Professor of Informatics; Education* (environmental informatics, educational technology, computer graphics/visualization/digital arts)

Gene Y. Tsudik, Ph.D. University of Southern California, *UCI Chancellor's Professor of Computer Science*

Kojiro Umezaki, M.A. Dartmouth College, *Associate Professor of Music; Computer Science*

Jessica Utts, Ph.D. Pennsylvania State University, *Department Chair and Professor of Statistics*

Joachim S. Vandekerckhove, Ph.D. University of Leuven, *Assistant Professor of Cognitive Sciences; Statistics* (response time modeling, model fitting, computational statistics, psychometrics, Bayesian statistics)

Alexander Veidenbaum, Ph.D. University of Illinois at Urbana-Champaign, *Professor of Computer Science*

Nalini Venkatasubramanian, Ph.D. University of Illinois at Urbana-Champaign, *Professor of Computer Science*

Alladi Venkatesh, Ph.D. Syracuse University, *Professor of Paul Merage School of Business; Informatics* (social impacts of information technology, Internet and the New Economy, Smart Home technologies, children and multimedia)

Richert Wang, Ph.D. University of California, Irvine, *Lecturer of Computer Science*

Mark J. Warschauer, Ph.D. University of Hawaii at Manoa, *Professor of Education; Informatics* (language, literacy, technology)

Xaiohui Xie, Ph.D. Massachusetts Institute of Technology, *Associate Professor of Computer Science; Developmental and Cell Biology* (computational biology, bioinformatics, genomics, neural computation, machine learning)

Guoqing Xu, Ph.D. Ohio State University, *Assistant Professor of Computer Science*

Yaming Yu, Ph.D. Harvard University, *Associate Professor of Statistics*

Zhaoxia Yu, Ph.D. William Marsh Rice University, *Associate Professor of Statistics*

Charles S. Zender, Ph.D. University of Colorado Boulder, *Professor of Earth System Science; Computer Science*

Hong-Kai Zhao, Ph.D. University of California, Los Angeles, *Professor of Mathematics; Computer Science* (applied and computational mathematics, inverse problems and imaging)

Shuang Zhao, Ph.D. Cornell University, *Assistant Professor of Computer Science* (computer graphics with a focus on material appearance modeling and physically-based rendering)

Hadar Ziv, Ph.D. University of California, Irvine, *Lecturer of Informatics* (software testing, requirements engineering, Bayesian modeling)

André W. van der Hoek, Ph.D. University of Colorado Boulder, *Professor of Informatics* (software engineering)