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.

Undergraduate Major in Informatics

Students in the Informatics major study human-computer interaction, social computing, computer-supported cooperative work, ubiquitous computing, organizational computing, and other topics that address the relationship between information technology design and use in social and organizational settings. The Informatics major addresses the broad set of issues surrounding design, all from a multidisciplinary perspective that includes computer science, information science, organizational science, social science, and cognitive science.

Courses offer extensive treatment of the conceptual underpinnings of the discipline and provide in-depth practical experiences, often performed on real-world examples and involving outside organizations. Students completing the major will be well suited for advanced careers in information technology or for further study at the graduate level. Specific careers include, but are not limited to, system or information analyst; system or information designer; project manager; and user interface and interaction designer. Career settings include corporations, nonprofit organizations, start-ups, and independent consulting.

Informatics majors complete one of two specializations: Human-Computer Interaction (HCI) or Organizations and Information Technology (OIT). More information is available online at http://www.ics.uci.edu/informatics/ugrad.

Admissions

Freshmen Applicants: See the Undergraduate Admissions (catalogue.uci.edu/previouseditions/2013-14/preadmissionmatters/undergraduateadmissions/#freshman) section.

Transfer Applicants:

Students transferring into the major must satisfy the following requirements:

1. Completion of one college-level mathematics course; courses equivalent to I&C SCI 6B Boolean Algebra and Logic, STATS 7 Basic Statistics or STATS 67 Introduction to Probability and Statistics for Computer Science are preferred as these courses facilitate scheduling after transfer to UCI.

2. Completion of one year of transferable computer science courses* with at least one course involving concepts such as those found in Java, Python, Scheme, C++, or other object-oriented or high-level programming language.

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.

More information is available at http://www.ics.uci.edu/informatics/ugrad or at the ICS Student Affairs Office; telephone (949) 824-5156; e-mail: ucounsel@uci.edu.

Requirements for the B.S. Degree in Informatics

All students must meet the University Requirements (catalogue.uci.edu/previouseditions/2013-14/informationforadmittedstudents/requirementsforabachelorsdegree).

Major Requirements

Lower-division

A. Select one of the following:

- IN4MATX 41-42-45 Informatics Core Course I and Informatics Core Course II and Patterns of Software Construction

- or

- I&C SCI 31-32-33 Introduction to Programming and Programming with Software Libraries and Intermediate Programming

or

- 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 21-22-IN4MATX 45 Introduction to Computer Science I and Introduction to Computer Science II and Patterns of Software Construction

B. Select one of the following:

- I&C SCI 90 New Students Seminar

- or IN4MATX 44 Seminar in Informatics Research Topics

C. Select one of the following:

- I&C SCI 52 Introduction to Software Engineering

- or IN4MATX 43 Introduction to Software Engineering

D. Complete:

- I&C SCI 45J Programming in Java as a Second Language

E. Complete:

- I&C SCI 6B Basic Statistics and

- or STATS 7 Introduction to Probability and Statistics for Computer Science

Upper-division
A. Informatics Core Requirements:

IN4MATX 113  Requirements Analysis and Engineering
IN4MATX 121  Software Design I
IN4MATX 131  Human Computer Interaction
IN4MATX 151  Project Management
IN4MATX 161  Social Analysis of Computerization
IN4MATX 191A-191B-191C  Senior Design Project and Senior Design Project

B. One of the following specializations:

Human-Computer Interaction

Complete:
IN4MATX 132  Project in Human-Computer Interaction Requirements and Evaluation

and select three of the following:
IN4MATX 133  User Interaction Software
IN4MATX 141  Information Retrieval
IN4MATX 143  Information Visualization
IN4MATX 153  Computer Supported Cooperative Work
IN4MATX 162W  Organizational Information Systems
IN4MATX 171  Introduction to Medical Informatics

and select two project courses from the following:
IN4MATX 125  Computer Game Development
IN4MATX 134  Project in User Interaction Software
IN4MATX 148  Project in Ubiquitous Computing
IN4MATX 163  Project in the Social and Organizational Impacts of Computing

and select four additional courses from the following:
Informatics 100–190
PUBHLTH 166  Geographic Information Systems

Organizations and Information Technology

Complete:
IN4MATX 141  Information Retrieval
IN4MATX 162W  Organizational Information Systems
IN4MATX 163  Project in the Social and Organizational Impacts of Computing

MGMT 5  Management of Contemporary Organizations

MGMT 102  Managing Organizational Behavior

and select four of the following:

MGMT 107  Introduction to Management Information Systems
MGMT 159  Design Management
MGMT 170  Technologies for Business
MGMT 173  Business Intelligence for Analytical Decisions
MGMT 175  Information Technology (IT) and Strategy

Sample Program of Study — Informatics: Human-Computer Interaction (HCI)

<table>
<thead>
<tr>
<th>Year</th>
<th>Fall</th>
<th>Winter</th>
<th>Spring</th>
</tr>
</thead>
<tbody>
<tr>
<td>Freshman</td>
<td>I&amp;C SCI 31</td>
<td>I&amp;C SCI 32</td>
<td>I&amp;C SCI 33</td>
</tr>
<tr>
<td></td>
<td>I&amp;C SCI 6B</td>
<td>WRITING 39A</td>
<td>WRITING 39C</td>
</tr>
<tr>
<td></td>
<td>STATS 7</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>WRITING 39A</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>WRITING 39B</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>WRITING 39C</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sophomore</td>
<td>I&amp;C SCI 45J</td>
<td>Informatics Core Specialization</td>
<td>Informatics Core Specialization</td>
</tr>
<tr>
<td></td>
<td>Informatics Core Specialization</td>
<td>GE III</td>
<td>GE III</td>
</tr>
<tr>
<td></td>
<td>GE III</td>
<td>GE IV</td>
<td></td>
</tr>
<tr>
<td></td>
<td>UD Writing</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Junior</td>
<td>Informatics Core Specialization</td>
<td>IN4MATX 191A</td>
<td></td>
</tr>
<tr>
<td></td>
<td>GE III</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Specialization</td>
<td>Specialization</td>
<td>Specialization</td>
</tr>
<tr>
<td></td>
<td>GE IV</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Senior</td>
<td>IN4MATX 191B</td>
<td>IN4MATX 191C</td>
<td>Elective</td>
</tr>
<tr>
<td></td>
<td>Specialization</td>
<td>Specialization</td>
<td>GE VIII</td>
</tr>
<tr>
<td></td>
<td>Specialization</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Sample Program of Study — Informatics: Organizations and Information Technology (OIT)

<table>
<thead>
<tr>
<th>Year</th>
<th>Fall</th>
<th>Winter</th>
<th>Spring</th>
</tr>
</thead>
<tbody>
<tr>
<td>Freshman</td>
<td>I&amp;C SCI 31</td>
<td>I&amp;C SCI 32</td>
<td>I&amp;C SCI 33</td>
</tr>
<tr>
<td></td>
<td>I&amp;C SCI 6B</td>
<td>WRITING 39A</td>
<td>WRITING 39C</td>
</tr>
<tr>
<td></td>
<td>STATS 7</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>WRITING 39A</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>WRITING 39B</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>WRITING 39C</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sophomore</td>
<td>I&amp;C SCI 45J</td>
<td>Informatics Core Specialization</td>
<td>Informatics Core Specialization</td>
</tr>
<tr>
<td></td>
<td>Informatics Core Specialization</td>
<td>GE III</td>
<td>GE III</td>
</tr>
<tr>
<td></td>
<td>GE III</td>
<td>GE IV</td>
<td></td>
</tr>
<tr>
<td></td>
<td>UD Writing</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Major and minor restrictions: Informatics majors pursuing minors outside of the Bren School of ICS may not count more than four courses toward both the major and minor.

Sample Program of Study — Informatics: Organizations and Information Technology (OIT)
Minor in Digital Information Systems

Students outside the Bren School of ICS may pursue a minor in Digital Information Systems (DIS). The minor is designed for students who want to learn about information systems, computation, and digital communication without preparing to be computer programmers. Students completing the DIS minor will be able to understand the role of digital information systems in society, and will learn about the technological underpinnings of these systems and constraints on their design and use.

Requirements for the Minor

Select two of the following:

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Course Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>I&amp;C SCI 3</td>
<td>Internet Technologies and their Social Impact</td>
</tr>
<tr>
<td>I&amp;C SCI 4</td>
<td>Human Factors for the Web</td>
</tr>
<tr>
<td>I&amp;C SCI 5</td>
<td>Environmental Issues in Information Technology</td>
</tr>
<tr>
<td>I&amp;C SCI 7</td>
<td>Introducing Modern Computational Tools</td>
</tr>
<tr>
<td>I&amp;C SCI 8</td>
<td>Practical Computer Security</td>
</tr>
<tr>
<td>I&amp;C SCI 11</td>
<td>The Internet and Public Policy</td>
</tr>
<tr>
<td>I&amp;C SCI 22</td>
<td>Introduction to Computer Science II</td>
</tr>
<tr>
<td>or CSE 22</td>
<td>Introduction to Computer Science II</td>
</tr>
<tr>
<td>I&amp;C SCI H22</td>
<td>Honors Introduction to Computer Science II</td>
</tr>
<tr>
<td>IN4MATX 42</td>
<td>Informatics Core Course II</td>
</tr>
<tr>
<td>or I&amp;C SCI 32</td>
<td>Programming with Software Libraries</td>
</tr>
<tr>
<td>IN4MATX 43</td>
<td>Introduction to Software Engineering</td>
</tr>
<tr>
<td>I&amp;C SCI 61</td>
<td>Game Systems and Design</td>
</tr>
</tbody>
</table>

Select one of the following:

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Course Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>I&amp;C SCI 10</td>
<td>How Computers Work</td>
</tr>
<tr>
<td>I&amp;C SCI 21</td>
<td>Introduction to Computer Science I</td>
</tr>
<tr>
<td>or CSE 21</td>
<td>Introduction to Computer Science I</td>
</tr>
<tr>
<td>I&amp;C SCI H21</td>
<td>Honors Introduction to Computer Science I</td>
</tr>
<tr>
<td>IN4MATX 41</td>
<td>Informatics Core Course I</td>
</tr>
<tr>
<td>or I&amp;C SCI 31</td>
<td>Introduction to Programming</td>
</tr>
</tbody>
</table>

Select four of the following:

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Course Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>I&amp;C SCI 105</td>
<td>Digital Information Systems</td>
</tr>
<tr>
<td>IN4MATX 131</td>
<td>Human Computer Interaction</td>
</tr>
<tr>
<td>IN4MATX 132</td>
<td>Project in Human-Computer Interaction Requirements and Evaluation</td>
</tr>
<tr>
<td>IN4MATX 143</td>
<td>Information Visualization</td>
</tr>
<tr>
<td>IN4MATX 148</td>
<td>Project in Ubiquitous Computing</td>
</tr>
<tr>
<td>IN4MATX 151</td>
<td>Project Management</td>
</tr>
<tr>
<td>IN4MATX 161</td>
<td>Social Analysis of Computerization</td>
</tr>
<tr>
<td>IN4MATX 162W</td>
<td>Organizational Information Systems</td>
</tr>
<tr>
<td>IN4MATX 163</td>
<td>Project in the Social and Organizational Impacts of Computing</td>
</tr>
<tr>
<td>IN4MATX 171</td>
<td>Introduction to Medical Informatics</td>
</tr>
<tr>
<td>IN4MATX 172</td>
<td>Project in Health Informatics</td>
</tr>
</tbody>
</table>

Students cannot take both IN4MATX 43 and I&C SCI 105.

NOTE: Bren School of ICS majors may not minor in Digital Information Systems. Courses used to complete the minor in Digital Information Systems may not also count toward the requirements for the Information and Computer Science minor or the Informatics minor.

Minor in Health Informatics

The minor in Health Informatics prepares students to understand the expanding role of information technology (IT) in health care and to participate in creating IT solutions to health care issues. It includes course work and fieldwork addressing a variety of health care IT settings. Students completing this minor will gain practical experience applying IT to serve the health care needs of communities and individuals.

Requirements for the Minor

Complete:

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Course Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>IN4MATX 171</td>
<td>Introduction to Medical Informatics</td>
</tr>
<tr>
<td>IN4MATX 172</td>
<td>Project in Health Informatics</td>
</tr>
</tbody>
</table>

Select two of the following:

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Course Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>I&amp;C SCI 4</td>
<td>Human Factors for the Web</td>
</tr>
<tr>
<td>I&amp;C SCI 7</td>
<td>Introducing Modern Computational Tools</td>
</tr>
<tr>
<td>I&amp;C SCI 10</td>
<td>How Computers Work</td>
</tr>
<tr>
<td>I&amp;C SCI 31</td>
<td>Introduction to Programming</td>
</tr>
<tr>
<td>I&amp;C SCI 32</td>
<td>Programming with Software Libraries</td>
</tr>
</tbody>
</table>

Select four of the following:

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Course Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>I&amp;C SCI 10</td>
<td>How Computers Work</td>
</tr>
<tr>
<td>I&amp;C SCI 21</td>
<td>Introduction to Computer Science I</td>
</tr>
<tr>
<td>or CSE 21</td>
<td>Introduction to Computer Science I</td>
</tr>
<tr>
<td>I&amp;C SCI H21</td>
<td>Honors Introduction to Computer Science I</td>
</tr>
<tr>
<td>IN4MATX 41</td>
<td>Informatics Core Course I</td>
</tr>
<tr>
<td>or I&amp;C SCI 31</td>
<td>Introduction to Programming</td>
</tr>
<tr>
<td>IN4MATX 105</td>
<td>Digital Information Systems</td>
</tr>
<tr>
<td>IN4MATX 131</td>
<td>Human Computer Interaction</td>
</tr>
<tr>
<td>IN4MATX 121</td>
<td>Software Design I</td>
</tr>
<tr>
<td>IN4MATX 123</td>
<td>Software Architecture</td>
</tr>
<tr>
<td>IN4MATX 131</td>
<td>Human Computer Interaction</td>
</tr>
<tr>
<td>IN4MATX 133</td>
<td>User Interaction Software</td>
</tr>
<tr>
<td>IN4MATX 143</td>
<td>Information Visualization</td>
</tr>
<tr>
<td>COMPSCI 111</td>
<td>Digital Image Processing</td>
</tr>
<tr>
<td>COMPSCI 121/IN4MATX 141</td>
<td>Information Retrieval</td>
</tr>
<tr>
<td>COMPSCI 122A</td>
<td>Introduction to Data Management</td>
</tr>
<tr>
<td>COMPSCI 131</td>
<td>Parallel and Distributed Computing</td>
</tr>
<tr>
<td>COMPSCI 134</td>
<td>Computer and Network Security</td>
</tr>
<tr>
<td>COMPSCI/CSE 145A</td>
<td>Embedded Computing Systems</td>
</tr>
<tr>
<td>COMPSCI 171</td>
<td>Introduction to Artificial Intelligence</td>
</tr>
<tr>
<td>COMPSCI 178</td>
<td>Machine Learning and Data-Mining</td>
</tr>
</tbody>
</table>
Requirements for the Minor

these relationships must be addressed in information and software design. Understanding the relationships among computers and people, and how fields or address specific interests. The minor particularly centers on sufficient flexibility to pursue courses that complement their major career, or academic pursuit in which information and software design is an integral part but is not the primary focus. The minor allows students to fill any gaps. The field of Software Engineering is concerned with the creation and analysis of the complex software systems that underlie modern society. Research in Software Engineering targets software artifacts and the people who create them. The field is large, and it encompasses engineering design research, i.e., the creation of new software artifacts with some desirable properties, as well as empirical research, i.e., the study of the effects that software development tools and methods have in the context of software development teams. Topics include software architectures, testing and debugging, software development tools, formal languages, requirements engineering, mining of large software-related data sources, reverse engineering, and development processes.

The M.S. degree in SE complements undergraduate knowledge in related fields with a solid framework for understanding the development of complex software systems.

Undergraduate Preparation for Admission. Typically, incoming students will have an undergraduate degree in computer science, though students may have an undergraduate degree in any field. Additionally they must have significant experience in software development. The ideal applicant is one who shows a considerable analytical depth in the practice of software development, typically gained from first-hand experience with large projects. Students admitted without a major in computer science, informatics, or equivalent will be expected to take undergraduate courses to fill any gaps.

Incoming students who already have a M.S. in Computer Science or closely related field may be exempted from (part of) the pre-candidacy course requirements by petition to the Graduate Dean, as filed by the student’s faculty advisor.

Program of Study for the Ph.D. Degree

Pre-Candidacy Course Requirements

Students must complete four software engineering core courses, six elective courses, and two quarters of seminars, literature survey, and individual study courses.

1. Software Engineering Core Courses:

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Course Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>IN4MATX 211</td>
<td>Software Engineering</td>
</tr>
<tr>
<td>IN4MATX 212</td>
<td>Analysis of Programming Languages</td>
</tr>
</tbody>
</table>

   Complete:

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Course Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>IN4MATX 131</td>
<td>Human Computer Interaction</td>
</tr>
</tbody>
</table>

   and at least two additional upper-division courses in Informatics

Before enrolling in any course for the Informatics minor, students should ensure that they meet its prerequisites. See the course prerequisites listed in the Catalogue or on the Informatics Web site at http://www.ics.uci.edu/ informatics/ugrad .

NOTE: A maximum of two courses can be taken Pass/Not Pass to satisfy the minor in Informatics. Students majoring in Information and Computer Science, Computer Science, or Computer Science and Engineering cannot minor in Informatics. Students who are considering a major in Informatics must complete the Informatics courses with a letter grade.

Minor in Informatics

The minor provides a focused study of Informatics to supplement a student’s major program of study and prepares students for a profession, career, or academic pursuit in which information and software design is an integral part but is not the primary focus. The minor allows students sufficient flexibility to pursue courses that complement their major field or address specific interests. The minor particularly centers on understanding the relationships among computers and people, and how these relationships must be addressed in information and software design.

Requirements for the Minor

Select one of the following:

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Course Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUR SCI 110W</td>
<td>Frameworks for Professional Nursing Practice</td>
</tr>
<tr>
<td>PUBHLTH 101</td>
<td>Introduction to Epidemiology</td>
</tr>
<tr>
<td>PUBHLTH 104</td>
<td>Analytic and Applied Epidemiology</td>
</tr>
<tr>
<td>PUBHLTH 122</td>
<td>Health Policy</td>
</tr>
<tr>
<td>PUBHLTH 124</td>
<td>Environmental and Public Health Policy</td>
</tr>
</tbody>
</table>

Select one of the following:

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Course Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>IN4MATX 151</td>
<td>Project Management</td>
</tr>
<tr>
<td>IN4MATX 161</td>
<td>Social Analysis of Computerization</td>
</tr>
<tr>
<td>IN4MATX 162</td>
<td>Social Analysis of Computerization</td>
</tr>
<tr>
<td>STATS 7</td>
<td>Basic Statistics</td>
</tr>
<tr>
<td>STATS 8</td>
<td>Introduction to Biological Statistics</td>
</tr>
<tr>
<td>STATS 67</td>
<td>Introduction to Probability and Statistics for Computer Science</td>
</tr>
</tbody>
</table>

1 This course may only be counted by majors outside of the Bren School of ICS.

NOTE: No more than one of these courses may be used to satisfy both the requirements of this minor and the requirements of the student’s major. A student must earn a grade of C or better in all courses used to satisfy the requirements of this minor.

Graduate Program in Software Engineering

The field of Software Engineering is concerned with the creation and analysis of the complex software systems that underlie modern society. Research in Software Engineering targets software artifacts and the people who create them. The field is large, and it encompasses engineering design research, i.e., the creation of new software artifacts with some desirable properties, as well as empirical research, i.e., the study of the effects that software development tools and methods have in the context of software development teams. Topics include software architectures, testing and debugging, software development tools, formal languages, requirements engineering, mining of large software-related data sources, reverse engineering, and development processes.

The Ph.D. degree in Software Engineering (SE) offers students opportunities for graduate study in the spectrum of intellectual activity in SE. The M.S. degree in SE complements undergraduate knowledge in related fields with a solid framework for understanding the development of complex software systems.

Undergraduate Preparation for Admission. Typically, incoming students will have an undergraduate degree in computer science, though students may have an undergraduate degree in any field. Additionally they must have significant experience in software development. The ideal applicant is one who shows a considerable analytical depth in the practice of software development, typically gained from first-hand experience with large projects. Students admitted without a major in computer science, informatics, or equivalent will be expected to take undergraduate courses to fill any gaps.

Incoming students who already have a M.S. in Computer Science or closely related field may be exempted from (part of) the pre-candidacy course requirements by petition to the Graduate Dean, as filed by the student’s faculty advisor.

Program of Study for the Ph.D. Degree

Pre-Candidacy Course Requirements

Students must complete four software engineering core courses, six elective courses, and two quarters of seminars, literature survey, and individual study courses.

1. Software Engineering Core Courses:

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Course Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>IN4MATX 211</td>
<td>Software Engineering</td>
</tr>
<tr>
<td>IN4MATX 212</td>
<td>Analysis of Programming Languages</td>
</tr>
</tbody>
</table>
3. Seminars and Individual Study:

- IN4MATX 209S: Seminar in Software (two quarters; four units each)
- IN4MATX 291S: Literature Survey in Software Engineering (two quarters; two units each)
- IN4MATX 299: Individual Study (two quarters; four units each)

2. Software Engineering Electives. Six elective courses chosen from the following courses offered by the School of ICS (all four units). The set of elective courses chosen by the student must be approved by the student’s research advisor. With the advisor’s permission, the student may substitute other non-seminar courses, as long as they are related to the student’s research interests.

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Course Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>IN4MATX 203</td>
<td>Qualitative Research Methods in Information Systems</td>
</tr>
<tr>
<td>IN4MATX 205</td>
<td>Quantitative Research Methods in Information Systems</td>
</tr>
<tr>
<td>IN4MATX 214</td>
<td>Requirements Engineering and Specification</td>
</tr>
<tr>
<td>IN4MATX 217</td>
<td>Software Processes</td>
</tr>
<tr>
<td>IN4MATX 219</td>
<td>Software Environments</td>
</tr>
<tr>
<td>IN4MATX 223</td>
<td>Applied Software Design Techniques</td>
</tr>
<tr>
<td>IN4MATX 231</td>
<td>User Interface Design and Evaluation</td>
</tr>
<tr>
<td>IN4MATX 233</td>
<td>Knowledge-Based User Interfaces</td>
</tr>
<tr>
<td>IN4MATX 235</td>
<td>Advanced User Interface Architecture</td>
</tr>
<tr>
<td>IN4MATX 241</td>
<td>Introduction to Ubiquitous Computing</td>
</tr>
<tr>
<td>IN4MATX 242</td>
<td>Ubiquitous Computing Interaction</td>
</tr>
<tr>
<td>IN4MATX 251</td>
<td>Computer-Supported Cooperative Work</td>
</tr>
<tr>
<td>IN4MATX 261</td>
<td>Social Analysis of Computing</td>
</tr>
<tr>
<td>IN4MATX 269</td>
<td>Computer Law</td>
</tr>
<tr>
<td>COMPSCI 203</td>
<td>Network and Distributed Systems Security</td>
</tr>
<tr>
<td>COMPSCI 221</td>
<td>Information Retrieval, Filtering, and Classification</td>
</tr>
<tr>
<td>COMPSCI 222</td>
<td>Principles of Data Management</td>
</tr>
<tr>
<td>COMPSCI 225</td>
<td>Next Generation Search Systems</td>
</tr>
<tr>
<td>COMPSCI 230</td>
<td>Distributed Computer Systems</td>
</tr>
<tr>
<td>COMPSCI 232</td>
<td>Computer and Communication Networks</td>
</tr>
<tr>
<td>COMPSCI 237</td>
<td>Middleware for Networked and Distributed Systems</td>
</tr>
<tr>
<td>COMPSCI 241</td>
<td>Advanced Compiler Construction</td>
</tr>
<tr>
<td>COMPSCI 273A</td>
<td>Machine Learning</td>
</tr>
<tr>
<td>COMPSCI 277</td>
<td>Data Mining</td>
</tr>
</tbody>
</table>

Qualifying Examinations

Written Comprehensive Examination

Students must pass a written examination testing their knowledge of the relevant topics and literature in Software Engineering and their ability to formulate clear arguments in writing and under time constraints. This examination is based on a predetermined reading list maintained by the program faculty. Preparation for this exam is done during two quarters of IN4MATX 291S. This exam is administered at most twice a year.

The exam is graded a Ph.D. PASS, M.S. PASS or FAIL. In case of M.S. PASS or FAIL, it may be re-taken once more, within 12 months, in an attempt to qualify for a Ph.D. PASS. A second M.S. PASS or FAIL results in disqualification of the student from the doctoral program (with or without a terminal M.S. degree).

Research Assessment

Students must find a faculty advisor and successfully complete a research project with that faculty member. The research project should be done over at least two quarters of independent study with that faculty member. The goal of this research assessment is to introduce the student to the practice of scientific publication.

Based on the project, the student must produce a research paper of publishable quality. This research paper must be reviewed by three faculty members in a peer-review process, revised by the student, and approved by the three faculty members.

The research assessment is graded PASS or FAIL. In case of FAIL, the student can re-submit the paper at most one more time within the maximum period of six months. A second FAIL results in disqualification from the program.

Advancement to Candidacy Examination

Each Ph.D. student must pass the oral advancement to candidacy exam, which assesses the student’s ability to conduct, present, and orally defend research work at the doctoral level. The research project and paper are the basis for the student’s oral advancement to candidacy exam. The oral candidacy exam consists of the research presentation by the student, followed by questions from the candidacy committee.

The student must complete the course requirements, and pass the two qualifying examinations prior to advancing to candidacy. The candidacy committee will consist of five faculty members, the majority of whom must be members of the student’s program, and is conducted in accordance with UCI Senate regulations.

Dissertation Topic Defense

The student must present a carefully articulated document representing the student’s dissertation plan. This document must include the proposed dissertation abstract, a discussion of the approach, a comprehensive survey of related work, and a plan for completing the work. The dissertation plan is presented by the student to the dissertation committee, who must unanimously approve the student’s proposal. The dissertation defense committee is formed in accordance to UCI Senate regulations.

Doctoral Dissertation and Final Examination

Students are required to complete a doctoral dissertation in accordance with Academic Senate regulations. In addition, they must pass an oral thesis defense which consists of a public presentation of the student’s research followed by an oral examination by the student’s doctoral committee. The committee must approve the thesis unanimously.
The normative time for advancement to candidacy is three years. The normative time for completion of the Ph.D. is six years, and the maximum time permitted is seven years.

Program of Study for the M.S. Degree

Course Requirements

M.S. students must complete four software engineering core courses, six elective courses, and two quarters of seminars (IN4MATX 298S). Students doing Capstone Plan I (Thesis) must complete two quarters, four units each, of Thesis Supervision (IN4MATX 298); students doing Capstone Plan II (Comprehensive Examination) must complete two quarters of literature survey courses.

The course requirements are identical to the Ph.D. degree, diverging only in making the Literature Survey and the Individual Study courses mutually exclusive, depending on the students’ Capstone option.

Capstone Requirement

Plan I: Thesis Option. Students must take and pass the Research Assessment examination. Additionally, students are required to defend their thesis in a public exam according to UCI Senate Policy. This requirement must be completed by the end of the second year.

Plan II: Comprehensive Examination Option. Students must take the written comprehensive examination, and obtain an M.S. PASS or higher. This requirement must be completed by the end of the second year. In case of FAIL, the exam may be re-taken once more. A second FAIL results in disqualification of the student from the master’s program.

Restriction

The M.S. degree will not be awarded to students who currently hold a M.S. degree in software engineering or a related field from the same or another university.

Requirements Beyond Graduate Division

Minimum Requirements

All Ph.D. students are expected to maintain a minimum GPA of 3.5 throughout the program. All M.S. students are expected to maintain a minimum GPA of 3.0 throughout the program. Failure to maintain this minimum will result in a recommendation that the student be disqualified. In addition, no grade lower than a B is counted toward satisfying any course requirements.

Graduate Concentrations

For graduate concentrations in Informatics, click here (catalogue.uci.edu/previouseditions/2013-14/donaldbrenschoolofinformationandcomputersciences/#graduatetext).

Faculty

Geoffrey Bowker: Values in design, social studies of databases, science and technology studies

Yunan Chen: Medical informatics, human-computer interaction

Paul Dourish: Human-computer interaction, computer-supported cooperative work

Daniel Frost: Artificial intelligence, software engineering, computer graphics, teaching of programming

Judith Gregory: Values in design, translational biomedical informatics, participatory design, design and emotion

Gillian Hayes: Interactive and collaborative technology, human-computer interaction, computer-supported cooperative work, educational technology, ubiquitous computing

Mizuko “Mimi” Ito: Ethnography, game studies, youth culture, learning sciences, online communities

James A. Jones: Software engineering, software testing and analysis, debugging and fault localization, static and dynamic analysis, software visualization

David G. Kay: Computer law, computer science education

Cory Knobel: Interactive and collaborative technology, values in design, modes of knowledge representation, philosophy of science and technology

Alfred Kobsa: User modeling, human-computer interaction, artificial intelligence, cognitive science, interdisciplinary computer science

Cristina Videira Lopes: Programming languages, acoustic communications, operating systems, software engineering

Gloria Mark: Computer-supported cooperative work, human-computer interaction

Melissa Mazmanian: Computer-mediated communication, organization studies, information and communication technologies in practice, social response to emerging technologies, work/non-work negotiations in the information age

Bonnie Nardi: Computer-supported collaborative work, human-computer interaction, computer-mediated communication, user studies methods, activity theory, cultural responses to technology development

Gary Olson: Interactive and collaborative technology, human-computer interaction, computer-supported cooperative work

Judy Olson: Interactive and collaborative technology, human-computer interaction, computer-supported cooperative work

Donald J. Patterson: Ubiquitous computing, pervasive computing, human-computer interaction, artificial intelligence, intelligent context for situated computing

Richard Pattis: Microworlds for teaching programming, debugging, computational tools for non-computer scientists

David F. Redmiles: Computer-supported cooperative work, human computer interaction, software engineering, globally distributed development teams, user interfaces, software tools

Debra J. Richardson: Software engineering; program testing; life-cycle validation; software environments

Thomas A. Standish: Software testing and analysis, software semantics and epistemology, programming and cognition, and software comprehension

Richard Taylor: Software engineering, user interfaces, environments, team support

Bill Tomlinson: Environmental informatics, educational technology, computer graphics/visualization/digital arts
The faculty in the Department of Informatics also contribute to the Informatics concentration in the ICS graduate program. This includes the General track, Interactive and Collaborative Technology track, and Ubiquitous Computing track.

Affiliated Faculty

Christopher Dobrian: Electronic music, composition
Magda El Zarki: Telecommunications, networks, wireless communication, video transmission
Tony Givargis: Embedded systems, platform-based system-on-a-chip design, low-power electronics
Vijay Gurbaxani: Economics of information systems management, impact of information technology on organization and market structure
Peter Krapp: Digital culture, media theory, cultural memory, history, and theory of artificial worlds
Robert Nideffer: Electronic intermedia, interface theory and design, technology and culture, contemporary social theory
Simon Penny: Robotic sculpture, interactive environments, electronic media, art practice history, and critical theory
Kavita Philip: Science and technology studies, South Asian studies, political ecology, critical studies of race, gender, colonialism, new media, and globalization
Alladi Venkatesh: Social impacts of information technology, Internet and the New Economy, Smart Home technologies, children and multimedia
Mark Warschauer: Language, literacy, technology

Courses

IN4MATX 41. Informatics Core Course I. 6 Units.
Fundamental concepts of computer software design and construction. Data, algorithms, functions, and abstractions. Overview of computer systems: data representation, architectural components, operating systems, networks. Introduction to information systems: parties involved, architectural alternatives, usability, organizational and social concerns.

Restriction: IN4MATX 41 may not be taken for credit if taken after I&C SCI 22 or CSE 22.

(II, Vb)

IN4MATX 42. Informatics Core Course II. 6 Units.
Alternative data structure implementations; analysis of time and space efficiency. Object-oriented programming concepts and techniques: classes, objects, inheritance, interfaces. Formal languages and automata. Problem modeling and design tradeoffs.

Prerequisite: IN4MATX 41. IN4MATX 41 with a grade of C or better.

Overlaps with I&C SCI 32, CSE 42, I&C SCI 33, CSE 43, I&C SCI 22, CSE 22.

(II, Vb)

IN4MATX 43. Introduction to Software Engineering. 4 Units.
Concepts, methods, and current practice of software engineering. Large-scale software production, software life cycle models, principles and techniques for each stage of development.

Overlaps with I&C SCI 52, I&C SCI 105.

IN4MATX 44. Seminar in Informatics Research Topics. 2 Units.
Introduction to current research topics in Informatics. Various faculty members present current research and relate it to the course content of the Informatics degree program.

Grading Option: Pass/no pass only.

IN4MATX 45. Patterns of Software Construction. 4 Units.
Building software applications; reusing and integrating components; designing for reuse. Effective use of libraries and APIs, file and network I/O, creation of user interfaces.

Prerequisite: IN4MATX 42 or I&C SCI 22 or CSE 22 or I&C SCI H22. I&C SCI H22 with a grade of C or better.

(Vb)

IN4MATX 101. Concepts in Programming Languages I. 4 Units.
In-depth study of several contemporary programming languages stressing variety in data structures, operations, notation, and control. Examination of different programming paradigms, such as logic programming, functional programming and object-oriented programming; implementation strategies, programming environments, and programming style. Course may be offered online.

Prerequisite: (IN4MATX 42 or I&C SCI 51 or CSE 31 or EECS 31) and (IN4MATX 45 or I&C SCI 23 or CSE 23 or I&C SCI 33 or CSE 43). IN4MATX 42 with a grade of C or better. I&C SCI 51 with a grade of C or better. CSE 31 with a grade of C or better. EECS31 with a grade of C or better. IN4MATX 45 with a grade of C or better. I&C SCI 23 with a grade of C or better. CSE 23 with a grade of C or better. I&C SCI 33 with a grade of C or better. CSE 43 with a grade of C or better.

Same as COMPSCI 141, CSE 141.

IN4MATX 102. Concepts of Programming Language II. 4 Units.
In-depth study of major programming paradigms: imperative, functional, declarative, object-oriented, and aspect-oriented. Understanding the role of programming languages in software development and the suitability of languages in context. Domain-specific languages. Designing new languages for better software development support.

Prerequisite: IN4MATX 101 or COMPSCI 141 or CSE 141. CSE 141 with a grade of C or better.
IN4MATX 113. Requirements Analysis and Engineering. 4 Units.

Aims to equip students to develop techniques of software-intensive systems through successful requirements analysis techniques and requirements engineering. Students learn systematic process of developing requirements through cooperative problem analysis, representation, and validation.

Prerequisite: IN4MATX 42 and (I&C SCI 22 or CSE 22 or I&C SCI 33 or CSE 43) and (IN4MATX 43 or I&C SCI 52) IN4MATX 42 with a grade of C or better. I&C SCI 22 with a grade of C or better. CSE 22 with a grade of C or better. I&C SCI 33 with a grade of C or better. CSE 43 with a grade of C or better. IN4MATX 43 with a grade of C or better. I&C SCI 52 with a grade of C or better.

IN4MATX 115. Software Testing, Analysis, and Quality Assurance. 4 Units.

Aims to prepare students to develop high-quality software through successful verification and validation techniques. Fundamental principles of software testing, how to test software, and how to ensure the thoroughness of testing to gain confidence in the correctness of the software.

Prerequisite: I&C SCI 6B and (IN4MATX 42 or I&C SCI 22 or CSE 22 or I&C SCI 33 or CSE 43) and (IN4MATX 43 or I&C SCI 52). CSE 43 with a grade of C or better. I&C SCI 52 with a grade of C or better.

IN4MATX 117. Project in Software System Design. 4 Units.

Specification, design, construction, testing, and documentation of a complete software system. Special emphasis on the need for and use of teamwork, careful planning, and other techniques for working with large systems.

Prerequisite: (IN4MATX 43 or I&C SCI 52) and (I&C SCI 33 or CSE 43 or I&C SCI 22 or CSE 22 or IN4MATX 42) I&C SCI 52 with a grade of a C or better. IN4MATX 42 with a grade of C or better.

Restriction: Upper-division students only.

IN4MATX 121. Software Design I. 4 Units.

Introduction to application design: designing the overall functionality of a software application. Topics include general design theory, software design theory, and software architecture. Includes practice in designing and case studies of existing designs.

Prerequisite: IN4MATX 45 or I&C SCI 23 or CSE 23 or I&C SCI 33 or CSE 43. IN4MATX 45 with a grade of C or better. I&C SCI 23 with a grade of C or better. CSE 23 with a grade of C or better. I&C SCI 33 with a grade of C or better. CSE 43 with a grade of C or better.

Restriction: Upper-division students only.

IN4MATX 122. Software Design II. 4 Units.

Introduction to implementation design: designing the internals of a software application. Topics include design aesthetics, design implementation, design recovery, design patterns, and component reuse. Includes practice in designing and case studies of existing designs.

Prerequisite: (I&C SCI 45J or I&C SCI 23 OR I&C SCI 46 OR IN4MATX 45) and (IN4MATX 101 or COMPSCI 141 or CSE 141).

IN4MATX 123. Software Architecture. 4 Units.

Prepares students to engineer well-structured software systems. Students learn a wide range of software architectural styles, architectural platforms that provide standard services to applications, and formal architecture description languages.

Prerequisite: IN4MATX 122 or ((IN4MATX 101 or COMPSCI 141 or CSE 141) and IN4MATX 113).

IN4MATX 124. Internet Applications Engineering. 4 Units.

Concepts in Internet applications engineering with emphasis on the Web. Peer-to-Peer and Interoperability. Topics include HTTP and REST, Remote Procedure/Method Calls, Web Services, data representations, content distribution networks, identity management, relevant W3C/IETF standards, and relevant new large-scale computing styles.

Prerequisite: COMPSCI 132.

Same as COMPSCI 137.

Restriction: Upper-division students only.

IN4MATX 125. Computer Game Development. 4 Units.

Introduction to the principles of interactive 2D and 3D computer game development. Concepts in computer graphics, algorithms, software engineering, art and graphics, music and sound, story analysis, and artificial intelligence are presented and are the basis for student work.

Prerequisite: COMPSCI 112 or COMPSCI 171 or IN4MATX 121 or ART 106B or I&C SCI 163 or I&C SCI 166.

Same as COMPSCI 113.

IN4MATX 131. Human Computer Interaction. 4 Units.

Basic principles of human-computer interaction (HCI). Introduces students to user interface design techniques, design guidelines, and usability testing. Students gain the ability to design and evaluate user interfaces and become familiar with some of the outstanding research problems in HCI.

Prerequisite: IN4MATX 41 or I&C SCI 10 or I&C SCI H21 or I&C SCI 21 or CSE 21 or I&C SCI H21 or I&C SCI 31 or CSE 41 or ENGR 10 or ENGRMAE 10 or EECS 10. IN4MATX 41 with a grade of C or better. I&C SCI 10 with a grade of C or better. I&C SCI 21 with a grade of C or better. CSE 21 with a grade of C or better. I&C SCI H21 with a grade of C or better. I&C SCI 31 with a grade of C or better. CSE 41 with a grade of C or better. ENGR 10 with a grade of C or better. EECS 10 with a grade of C or better.

Restriction: Upper-division students only.

IN4MATX 132. Project in Human-Computer Interaction Requirements and Evaluation. 4 Units.

Students undertake significant projects in the elicitation and specification of HCI requirements and the thorough evaluation of user interfaces.

Prerequisite: IN4MATX 131.
IN4MATX 133. User Interaction Software. 4 Units.
Prerequisite: IN4MATX 45 or I&C SCI 23 or CSE 23 or ((I&C SCI 33 or CSE 43) and I&C SCI 45J). IN4MATX 45 with a grade of C or better. I&C SCI 23 with a grade of C or better. CSE 23 with a grade of C or better. I&C SCI 33 with a grade of C or better. CSE 43 with a grade of C or better. I&C SCI 45J with a grade of C or better.

IN4MATX 134. Project in User Interaction Software. 4 Units.
Students complete an end-to-end user interface programming project based on an iterative design paradigm. Topics may include requirements brainstorming, paper prototyping, iterative development, cognitive walkthrough, quantitative evaluation, and acceptance testing. Materials fee.
Prerequisite: IN4MATX 131 and IN4MATX 133.

IN4MATX 141. Information Retrieval. 4 Units.
An introduction to information retrieval including indexing, retrieval, classifying, and clustering text and multimedia documents.
Prerequisite: (IN4MATX 45 or I&C SCI 46 or CSE 46 or ((I&C SCI 33 or CSE 43) and I&C SCI 45J)) and (STATS 7 or STATS 67). IN4MATX 45 with a grade of C or better. I&C SCI 46 with a grade of C or better. CSE 46 with a grade of C or better. I&C SCI 33 with a grade of C or better. CSE 43 with a grade of C or better. I&C SCI 45J with a grade of C or better.
Same as COMPSCI 121.

IN4MATX 143. Information Visualization. 4 Units.
Introduction to interactive visual interfaces for large datasets, and to principles of human visual perception and human computer interaction that inform their design. Various applications for data analysis and monitoring are discussed.
Prerequisite: IN4MATX 131 or I&C SCI 52 or (IN4MATX 43 and (I&C SCI 31 or CSE 41 or I&C SCI 21 or CSE 21 or IN4MATX 41)). IN4MATX 131 with a grade of C or better. I&C SCI 52 with a grade of C or better. IN4MATX 43 with a grade of C or better. I&C SCI 31 with a grade of C or better. CSE 41 with a grade of C or better. I&C SCI 21 with a grade of C or better. CSE 21 with a grade of C or better. IN4MATX 41 with a grade of C or better.

IN4MATX 148. Project in Ubiquitous Computing. 4 Units.
Introduction to ubiquitous computing research methods, tools, and techniques. Prototyping, design, and evaluation of physical computing applications, smart environments, embedded systems, and future computing scenarios. Includes hands-on in-class laboratory exercises. Materials fee.
Prerequisite: I&C SCI 10 or I&C SCI 21 or I&C SCI 31 or IN4MATX 41. IN4MATX 41 with a grade of C or better.
Restriction: Upper-division students only.

IN4MATX 151. Project Management. 4 Units.
Introduces theoretical and practical aspects of project management. Topics include organizational theory, group behavior, project management skills, case studies, personal and group productivity tools, management of distributed work, stakeholders, consultants, and knowledge management. Students do a project exercise.
Prerequisite: IN4MATX 43 or I&C SCI 52. I&C SCI 52 with a grade of C or better.
Restriction: Upper-division students only.

IN4MATX 153. Computer Supported Cooperative Work. 4 Units.
Introduces concepts and principles of collaborative systems. Topics may include shared workspaces, group interaction, workflow, architectures, interaction between social and technical features of group work, and examples of collaborative systems used in real-world settings. Students develop a simple collaborative application.
Prerequisite: (IN4MATX 161 or I&C SCI 52 or IN4MATX 43) and (I&C SCI 31 or CSE 41 or I&C SCI 21 or CSE 21 or IN4MATX 41). I&C SCI 52 with a grade of C or better. I&C SCI 31 with a grade of C or better. CSE 41 with a grade of C or better. I&C SCI 21 with a grade of C or better. CSE 21 with a grade of C or better. IN4MATX 41 with a grade of C or better.

IN4MATX 161. Social Analysis of Computerization. 4 Units.
Introduction of computerization as a social process. Examines the social opportunities and problems raised by new information technologies, and the consequences of different ways of organizing. Topics include computerization and work life, privacy, virtual communities, productivity paradox, systems risks.
Prerequisite: Satisfactory completion of the Lower-Division Writing requirement. IN4MATX 41 or I&C SCI 10 or I&C SCI 21 or CSE 21 or I&C SCI 31 or CSE 41 or ENGR 10 or EECS 10 or ENGRMAE 10, IN4MATX 41 with a grade of C or better. I&C SCI 10 with a grade of C or better. I&C SCI 21 with a grade of C or better. CSE 21 with a grade of C or better. I&C SCI 31 with a grade of C or better. CSE 41 with a grade of C or better. ENGR 10 with a grade of C or better. EECS 10 with a grade of C or better. ENGRMAE 10 with a grade of C or better.

IN4MATX 162W. Organizational Information Systems. 4 Units.
Introduction to role of information systems in organizations, components and structure of organizational information systems, and techniques used in information systems analysis, design, and implementation.
Prerequisite: IN4MATX 161. Satisfactory completion of the Lower-Division Writing requirement.

IN4MATX 163. Project in the Social and Organizational Impacts of Computing. 4 Units.
Students undertake projects intended to gather and analyze data from situations in which computers are used, organize and conduct experiments intended to test hypotheses about impacts, and explore the application of concepts learned in previous courses.
Prerequisite: IN4MATX 162.
IN4MATX 171. Introduction to Medical Informatics. 4 Units.
Broad overview of medical informatics for students with varied backgrounds. Electronic medical records, online resources, mobile technologies, patient safety, and computational design. Legal, ethical, and public policy issues. Health systems management. Evaluation and fieldwork for health systems.

Same as PUBHLTH 105.

Restriction: Upper-division students only.

IN4MATX 172. Project in Health Informatics. 4 Units.
Students undertake significant quarter-long projects related to health informatics. Topics may include field evaluations of health care technologies, prototypes, iterative design, and system implementations.

Prerequisite: PUBHLTH 105 or IN4MATX 171.

Same as PUBHLTH 106.

IN4MATX 190. Special Topics in Informatics. 4 Units.
Studies in selected areas of informatics. Topics addressed vary each quarter.

Prerequisite: Prerequisites vary.

Repeatability: Unlimited as topics vary.

IN4MATX 191A. Senior Design Project. 4 Units.
Group supervised project in which students analyze, specify, design, construct, evaluate, and adapt a significant information processing system. Topics include team management, professional ethics, and systems analysis.

Corequisite: IN4MATX 123.
Prerequisite: IN4MATX 121 and IN4MATX 131 and IN4MATX 151 and IN4MATX 161.

Restriction: Upper-division students only.

IN4MATX 191B. Senior Design Project. 4 Units.
Group supervised project in which students analyze, specify, design, construct, evaluate, and adapt a significant information processing system. Topics include team management, professional ethics, and systems analysis.

Prerequisite: IN4MATX 191A.

Grading Option: In progress only.

IN4MATX 191C. Senior Design Project. 4 Units.
Group supervised project in which students analyze, specify, design, construct, evaluate, and adapt a significant information processing system. Topics include team management, professional ethics, and systems analysis.

Prerequisite: IN4MATX 191B.

IN4MATX H198. Honors Research. 4 Units.
Directed independent research in Informatics for honors students.

Prerequisite: Satisfactory completion of the Lower-Division Writing requirement.

Repeatability: May be repeated for credit unlimited times.

Restriction: Bren School of ICS Honors Program or the Campuswide Honors Program students only.

IN4MATX 199. Individual Study. 2-5 Units.
Individual research or investigation under the direction of an individual faculty member.

Repeatability: May be repeated for credit unlimited times.

IN4MATX 201. Research Methodology for Informatics. 4 Units.
Introduction to strategies and idioms of research in Informatics. Includes examination of issues in scientific inquiry, qualitative and quantitative methods, and research design. Both classic texts and contemporary research literature are read and analyzed.

IN4MATX 203. Qualitative Research Methods in Information Systems. 4 Units.
Introduction to qualitative research methods used to study computerization and information systems, such as open-ended interviewing, participant observation, and ethnography. Studies of the methods in practice through examination of research literature.

Prerequisite: IN4MATX 261 or IN4MATX 251.

IN4MATX 205. Quantitative Research Methods in Information Systems. 4 Units.
Quantitative research methods used to study computerization and information systems. Design of instruments, sampling, sample sizes, and data analysis. Validity and reliability. Longitudinal versus cross-sectional designs. Analysis of secondary data. Studies of the methods through examination of research literature.

Prerequisite: (IN4MATX 251 or IN4MATX 261) and basic knowledge of elementary statistics.

IN4MATX 207S. Doctoral Seminar on Research and Writing. 2 Units.
Doctoral seminar centered on original research and writing. Provides a chance for doctoral students at all levels to present original work, brainstorm ongoing issues, and learn to provide and receive critical feedback from peers.

Grading Option: Satisfactory/unsatisfactory only.

Repeatability: May be repeated for credit unlimited times.

IN4MATX 209S. Seminar in Software. 2 Units.
Current research and research trends in informatics. Forum for presentation and criticism by students of research work in progress.

Repeatability: Unlimited as topics vary.

IN4MATX 211. Software Engineering. 4 Units.
Study of the concepts, methods, and tools for the analysis, design, construction, and measurement of complex software-intensive systems. Underlying principles emphasized. State-of-the-art software engineering and promising research areas covered, including project management.
IN4MATX 212. Analysis of Programming Languages. 4 Units.
Concepts in modern programming languages, their interaction, and the relationship between programming languages and methods for large-scale, extensible software development. Empirical analysis of programming language usage.

Same as COMPSCI 253.

IN4MATX 214. Requirements Engineering and Specification. 4 Units.
Study of rigorous techniques in requirements engineering – requirements definition phase of software development – with focus on modeling and specification. Topics include notations and models for requirements specification; and methods, tools and processes for software requirements elicitation, representation, analysis.

Restriction: Graduate students only.

IN4MATX 215. Software Analysis and Testing. 4 Units.
Studies techniques for developing confidence in software from traditional testing schemes to integrated, multitechnique analytic approaches. Considers strengths and weaknesses and explores opportunities for synergistic technique application. Emphasis is on approaches integrated into the software process.

IN4MATX 217. Software Processes. 4 Units.
Explores vehicles for modeling, coding, and analyzing software processes. Considers integration of software process programming approaches with traditional management issues. Explores the use of software process execution as a vehicle for effective integration of tools into environments.

IN4MATX 219. Software Environments. 4 Units.
Study of the requirements, concepts, and architectures of comprehensive, integrated, software development and maintenance environments. Major topics include process support, object management, communication, interoperability, measurement, analysis, and user interfaces in the environment context.

IN4MATX 221. Software Architecture. 4 Units.
Study of the concepts, representation techniques, development methods, and tools for architecture-centric software engineering. Topics include domain-specific software architectures, architectural styles, architecture description languages, software connectors, and dynamism in architectures.

IN4MATX 223. Applied Software Design Techniques. 4 Units.
Study of concepts, representations, techniques, and case studies in structuring software systems, with an emphasis on design considerations. Topics include static and dynamic system structure, data models, abstractions, naming, protocols and application programmer interfaces.

IN4MATX 225. Information Retrieval, Filtering, and Classification. 4 Units.
Algorithms for the storage, retrieval, filtering, and classification of textual and multimedia data. The vector space model, Boolean and probabilistic queries, and relevance feedback. Latent semantic indexing; collaborative filtering; and relationship to machine learning methods.

Prerequisite: COMPSCI 161 and COMPSCI 171 and (MATH 3A or MATH 6G).

Same as COMPSCI 221.

IN4MATX 231. User Interface Design and Evaluation. 4 Units.
Introduction to the design and evaluation of user interfaces, with an emphasis on methodology. Cognitive principles, design life cycle, on-line and off-line prototyping techniques. Toolkits and architectures for interactive systems. Evaluation techniques, including heuristic and laboratory methods.

IN4MATX 232. Research in Human-Centered Computing. 4 Units.
Introduction to contemporary topics in human-computer interaction, including methods, technologies, design, and evaluation. Emerging application domains and their challenges to traditional research methods. Advanced architectures and technologies. Critical issues.

Prerequisite: Some familiarity with HCI principles.

IN4MATX 233. Knowledge-Based User Interfaces. 4 Units.
Concepts related to the development of interactive software systems with a focus on knowledge-based tools and human-centered design. Topics span the fields of human-computer interaction, software engineering, and knowledge representation.

Prerequisite: COMPSCI 171.

IN4MATX 235. Advanced User Interface Architecture. 4 Units.
Architectural concerns in advanced interactive systems. The design of current and emerging platforms for novel interactive systems. Paradigms such as constraint-based programming, multimodal interaction, and perceptual user interfaces for individual, distributed, and ubiquitous applications.

IN4MATX 241. Introduction to Ubiquitous Computing. 4 Units.
The "disappearing computer" paradigm. Differences to the desktop computing model: applications, interaction in augmented environments, security, alternate media, small operating systems, sensors, and embedded systems design. Evaluation by project work and class participation.

Same as COMPSCI 248A.

IN4MATX 242. Ubiquitous Computing and Interaction. 4 Units.
Principles and design techniques for ubiquitous computing applications. Conceptual basis for tangible and embodied interaction. Interaction in virtual and augmented environments. Design methods and techniques. Design case studies. Examination by project work.

Prerequisite: IN4MATX 231 and IN4MATX 241.

Same as COMPSCI 248B.

IN4MATX 244. Introduction to Embedded and Ubiquitous Systems. 4 Units.
Embedded and ubiquitous system technologies including processors, DSP, memory, and software. System interfacing basics; communication strategies; sensors and actuators, mobile and wireless technology. Using pre-designed hardware and software components. Design case studies in wireless, multimedia, and/or networking domains.

Prerequisite: (I&C SCI 51 and COMPSCI 152 and (MATH 3A or MATH 6G or I&C SCI 6D) and COMPS1 161) or B.S. degree in Computer Science.

Same as COMPSCI 244.
IN4MATX 251. Computer-Supported Cooperative Work. 4 Units.
The role of information systems in supporting work in groups and organizations. Examines various technologies designed to support communication, information sharing, and coordination. Focuses on behavioral and social aspects of designing and using group support technologies.

IN4MATX 256. Social Analysis of Computing. 4 Units.
The social and economic impacts of computing and information technologies on groups, organizations, and society. Topics include computerization and changes in the character of work, social control and privacy, electronic communities, and risks of safety-critical systems to people.

IN4MATX 257. Computerization, Work, and Organizations. 4 Units.
Selected topics in the influence of computerization and information systems in transforming work and organizations. Theories of organization and organizational change. Processes by which diverse information technologies influence changes in work and organizations over short and long time periods.
Prerequisite: IN4MATX 251 or IN4MATX 251.

IN4MATX 258. Theories of Computerization and Information Systems. 4 Units.
Social and economic conceptions of information technology. Macrosocial and economic conditions that foster changes in information technologies. Social construction of information and computer technology in professional worlds. Theories of information technology and large-scale social change.
Prerequisite: IN4MATX 251 or IN4MATX 261.

IN4MATX 259. Digital Media and Society. 4 Units.
Selected topics in the technological and social aspects of online interactions, and policy including online games, social media, electronic activism, e-commerce, and digital libraries. Media-theoretic approaches to digital technology. Architectures, infrastructure considerations, and their consequences.
Prerequisite: IN4MATX 251 or IN4MATX 261.

IN4MATX 260. Computer Law. 4 Units.
Restriction: Graduate students only.

IN4MATX 261. Information Technology in Global Sustainability. 4 Units.
Explores the relationship between recent developments in information technology and the global transition to sustainability. Topics include the role of IT systems in the provision of human needs and wants (e.g., smart grids, food systems, and other IT-enabled infrastructure).
Restriction: Graduate students only.

IN4MATX 262. Research Seminar. 2 Units.
Forum for presentation and criticism by students of research work in progress. Presentation of problem areas and related work. Specific goals and progress of research.
Grading Option: Satisfactory/unsatisfactory only.
Repeatability: May be repeated for credit unlimited times.
Restriction: Graduate students only.

IN4MATX 263. Literature Survey in Software Engineering. 2 Units.
Reading and analysis of relevant literature in Software Engineering under the direction of a faculty member.
Repeatability: May be repeated for credit unlimited times.

IN4MATX 264. Special Topics in Informatics. 4 Units.
Studies in selected areas of informatics. Topics addressed vary each quarter.
Repeatability: Unlimited as topics vary.
Restriction: Graduate students only.

IN4MATX 265. Thesis Supervision. 2-12 Units.
Individual research or investigation conducted in preparation for the M.S. thesis option or the dissertation requirements for the Ph.D. program.
Repeatability: May be repeated for credit unlimited times.
Restriction: Graduate students only.

IN4MATX 266. Individual Study. 2-12 Units.
Individual research or investigation under the direction of an individual faculty member.
Repeatability: May be repeated for credit unlimited times.
Restriction: Graduate students only.