COMPUTER SCIENCE DEPARTMENT

Advisement Bulletin
2019-2020

Table of Contents

UNIVERSITY OF WISCONSIN OSHKOSH
Computer Science Department
Halsey 229
Phone (920) 424-2068
Fax (920) 424-0045
Website: http://www.uwosh.edu/computer_science

Faculty

David Furcy   Professor, Ph.D.
                                               Georgia Institute of Technology
                                               furcyd@uwosh.edu

Hannah Hillberg   Assistant Professor, Ph.D.
                                               University of Minnesota
                                               hillbergh@uwosh.edu

Erik Krohn   Associate Professor, Ph.D.
                                               University of Iowa
                                               krohne@uwosh.edu

Kevin Kurek (UWO-FDL)   Associate Lecturer
                                               kurek@uwosh.edu

Kathy Lynch   Senior Lecturer, M.S.
                                               Texas A&M University
                                               lynchk@uwosh.edu

Ahilan Sivasamy (UWO-Fox Cities)   Senior Lecturer, M.Sc.
                                               University of Bristol
                                               sivasamya@uwosh.edu

Scott Summers   Associate Professor, Ph.D.
                                               Iowa State University
                                               summerss@uwosh.edu

George Thomas, Chair   Associate Professor, Ph.D.
                                               University of Iowa
                                               thomasg@uwosh.edu

Admission & Graduation Requirements

- **To be eligible to declare a Computer Science Major**, students must be currently enrolled in or have completed Computer Science 221 Object-Oriented Design and Programming I.

- **To be eligible to declare a Software Technology Major**, students must have completed a Software Developer Associate in Applied Science (A.A.S.) degree from a Wisconsin technical college with which UW-Oshkosh has the appropriate articulation agreement to pursue their studies at UW-Oshkosh to complete a Bachelor of Science (B.S.) in Software Technology degree.

To be eligible for graduation, students must meet all requirements for the degree being sought in addition to earning a minimum grade point average of 2.00 in all courses required for the major or minor.

In addition to University and College requirements, students must have a minimum grade point average of 2.00 in those computer science (CS) courses numbered 300 or above that are used to satisfy the requirements of the major excluding CS 399 and CS 490.

Please note that this statement implies that if a student completes more than the minimum number of courses required for a specific emphasis in a major, the student may designate those courses that will be used for computing the grade point average when more than one choice is possible. The resulting set of designated courses must satisfy the requirements of a specific emphasis of that major.

Objectives

Three to five years after graduation, our typical alumni will:

1. Advance beyond initial entry-level positions as computing professionals or have made significant progress toward a graduate degree in computing.

2. Use the background they have acquired in a wide range of areas in computer science during their undergraduate study as a basis for continued growth of their professional knowledge and skills.

3. Use teamwork skills effectively in the development of complex software systems.

4. Use communication skills to advantage within the organizational structure of workplaces that are becoming increasingly diverse and interconnected among different groups including, but not necessarily limited to, those groups based on racial, gender, age, and religious backgrounds.

5. Demonstrate strong professional ethics in all of their computing endeavors.

Learning Outcomes

General outcomes that guide the specific learning outcomes for individual courses.

- Outcome A: An ability to apply knowledge of computing and mathematics appropriate to the discipline

- Outcome B: An ability to analyze a problem, and identify and define the computing requirements appropriate to its solution

- Outcome C: An ability to design, implement and evaluate a computer-based system, process, component or program to meet desired needs

- Outcome D: An ability to function effectively on teams to accomplish a common goal

- Outcome E: An understanding of professional, ethical, legal, security and social issues and responsibilities

- Outcome F: An ability to communicate effectively with a range of audiences

- Outcome G: An ability to analyze the local and global impact of computing on individuals, organizations and society

- Outcome H: Recognition of the need for and an ability to engage in continuing professional development

- Outcome I: An ability to use current techniques, skills and tools necessary for computing practice

- Outcome J: An ability to apply mathematical foundations, algorithmic principles and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the trade-offs involved in design choices

- Outcome K: An ability to apply design and development principles in the construction of software systems of varying complexity

Prerequisites and Repeating Courses

<u>Prerequisites</u>

The computer science department has established prerequisites for each course so that students will be adequately prepared for the topics covered in the course. This allows the instructor to proceed with the course in a manner that ensures quality in the program.

The department strictly enforces the prerequisite policy. You should not take courses unless you satisfy the prerequisites. Otherwise, when the deficiency is detected, you will have to drop the course. All courses are checked.

<u>Repeating Courses</u>

The College of Letters and Science and the computer science department permit students to repeat a given computer science course one time only. Note that this policy includes transfer courses. A very important consequence of this policy is that a student who does not earn a grade of C or better in certain courses that are prerequisites for a required course must withdraw from the appropriate major. For example, suppose a student earns a D grade in CS 271 (Data Structures). Then the student repeats it and earns a C- grade. Because CS 271 is a prerequisite for most 300-level courses, this student will not be permitted to major in Computer Science.

.

Computer Science Major

Core Courses

These courses are required of all Computer Science majors and emphases within the Computer Science major.

Course	Credits
CS 212 Discrete Structures	3 cr.
CS 221 Object Oriented Design and Programming I	3 cr.
CS 251 Computer Architecture and Assembly Language	3 cr.
CS 262 Object Oriented Design and Programming II	4 cr.
CS 271 Data Structures	4 cr.
CS 321 Algorithms	3 cr.
CS 331 Programming Languages	3 cr.
CS 341 Software Engineering I	3 cr.
PHIL 350 Computing Ethics	3 cr.
IS 208 Professional Career Skills in Math and Natural Science	1 cr.

One of the following:

CS 399 Internship in Computer Science	3 cr.
CS 490 Practicum in Computer Science	3 cr.

Total: 33 cr.

Computer Science Major

The Computer Science major focuses on knowledge and skills for the design, development, maintenance, and management of software, including software for the operation of instrumentation. It is appropriate for students preparing for careers as systems programmers, systems analysts, software engineers, or computer-based instrumentation specialists.

Requirements in addition to the computer science core courses:

Course	Credits
Either Math 171 Calculus I or Math 206 Mathematics for Business Analysis II	4-5 cr.
Either Math 201 Applied Statistics, Math 301 Introduction to Probability and Statistics, or Econ 210 Economic and Business Statistics	3 cr.
Electives: 12 credits from the elective courses list.	12 cr.
Total	**19-20 cr.**

Computer Science Major - Systems and Theory Emphasis

This emphasis focuses on the fundamental practical and theoretical foundations of computer science, providing an in-depth understanding of both the software and hardware components of computer systems. It is appropriate for students preparing for careers in scientific computing or research, and for students planning to pursue graduate degrees.

This emphasis is accredited by the Computing Accreditation Commission of ABET, 111 Market Place, Suite 150, Baltimore, MD 21202-4012, telephone: (410) 347-7700.

Requirements in addition to the computer science core courses:

Course	Credits
CS 310 Computer Organization and Design	3 cr.
CS 381 Theory of Computing	3 cr.
CS 421 Operating Systems	3 cr.
CS 431 Compilers	3 cr.
CS 499 Computer Science Assessment	0 cr.
Math 171 Calculus I	5 cr.
Math 172 Calculus II or Math 256 Linear Mathematics	3-4 cr.
Math 201 Applied Statistics or Math 301 Introduction to Probability and Statistics	3 cr.
Electives: 6 credits from the elective courses list. The chosen electives must not appear above as requirements for the computer science emphasis.	6 cr.

Total   29-30 cr.

Important Notes:

1. The Systems and Theory (ST) emphasis is open only to students pursuing the B.S. degree.

2. The math and science requirements of the ST emphasis necessitate completion of Mathematics 171, 172 or 256, 201 or 301; CS 212, 381; plus an additional 15 units (credits) from other math and science courses. In selecting courses for these additional 15 units (credits), science courses must be chosen to ensure the student meets the general university natural science requirement for a B.S. degree, and math courses must be numbered 222 or above.

3. CS 499 is open only to ST emphasis students who are in their final semester. Enrolled students must take the Major Field Test in Computer Science.

Computer Science Major - Elective Courses

1. CS 300 Artificial Intelligence

2. CS 310 Computer Organization and Design

3. CS 326 Computer Security

4. CS 342 Software Engineering II

5. CS 344 Mobile Application Development

6. CS 346 Web Software Development

7. CS 361 Database Systems

8. CS 371 Computer Graphics

9. CS 381 Theory of Computing

10. CS 391 Data Communication and Computer Networks

11. CS 421 Operating Systems

12. CS 431 Compilers

13. CS 474 Honors: Thesis (up to 3 credits)

14. CS 480 Topics in CS

Computer Science Minor

The Common Core (14 credits)

These courses are required of all computer science minors.

Course	Credits
CS 221 Object Oriented Design and Programming I	3 cr.
CS 251 Computer Architecture and Assembly Language	3 cr.
CS 262 Object Oriented Design and Programming II	4 cr.
CS 271 Data Structures	4 cr.
Total	**14 cr.**

Elective Computer Science Courses (9 credits)

Any **three** courses may be selected from the following upper-division offerings.

1. CS 300 Artificial Intelligence
2. CS 310 Computer Organization and Design
3. CS 321 Algorithms
4. CS 326 Computer Security
5. CS 331 Programming Languages
6. CS 341 Software Engineering I
7. CS 342 Software Engineering II
8. CS 344 Mobile Application Development
9. CS 346 Web Software Development
10. CS 361 Database Systems
11. CS 371 Computer Graphics
12. CS 381 Theory of Computing
13. CS 391 Computer Networks
14. CS 421 Operating Systems
15. CS 431 Compilers
16. CS 480 Topics in CS
17. CS 490 Practicum in Computer Science

Total: 23 cr.

Software Technology Major

Course	Credits
CS 221 Object Oriented Design and Programming I	3 cr.
CS 247 Introduction to Usability	3 cr.
CS 262 Object Oriented Design and Programming II	4 cr.
CS 271 Data Structures	4 cr.
CS 341 Software Engineering I	3 cr.
CS 344 Mobile Application Development	3 cr.
CS 346 Web Software Development	3 cr.
CS 399 Internship in Computer Science	3 cr.
PHIL 350 Computing Ethics	3 cr.
IS 208 Professional Career Skills in Math and Natural Science	1 cr.
INFO SYS 315 Database Systems in Business	3 cr.
INFO SYS 316 Networking and Data Communication	3 cr.

One of the following:

INFO SYS 318 Agile Application Development	3 cr.
INFO SYS 432 Agile Web Design and Development	3 cr.

Total: 39 cr.

Courses
Course Rotation Schedule – Oshkosh campus

The computer science department will attempt to offer courses according to the following rotation. **The schedule could be altered due to lack of resources or low enrollments.**

The following courses will be offered every semester:
CS 125 Web Site Development (Fall or Fall Interim, Spring or Spring Interim)
CS 142 Introduction to Computer Science and Programming
CS 212 Discrete Structures
CS 221 Object Oriented Design and Programming I
CS 251 Computer Architecture and Assembly Language
CS 262 Object Oriented Design and Programming II
CS 271 Data Structures

The following courses will be offered once per year:

Fall Semester	Spring Semester
CS 247 Introduction to Usability	CS 331 Programming Languages
CS 321 Algorithms	CS 361 Database Systems
CS 341 Software Engineering I	
CS 346 Web Software Development	

The following courses will be offered every third semester

Fall 2019
CS 344 Mobile Application Development
CS 391 Data Communications and Computer Networks
CS 431 Compilers

Spring 2020
CS 300 Artificial Intelligence
CS 326 Computer Security
CS 381 Theory of Computing

Fall 2020
CS 310 Computer Organization and Design
CS 371 Computer Graphics
CS 421 Operating Systems

Spring 2021
CS 344 Mobile Application Development
CS 391 Data Communications and Computer Networks
CS 431 Compilers

Course Descriptions

Courses

100 Level: 105, 120, 125, 142

200 Level: 212, 216, 221, 247, 251, 262, 271

300 Level: 300, 310, 321, 326, 331, 341, 342, 344, 346, 361, 371, 381, 391, 399

400 Level: 421, 431, 474, 480, 490

CS 105 Computer Applications (3 units)
Principles and use of computer applications including word processors, spreadsheets, and data bases. May also cover other applications such as telecommunications, graphics, statistics, simulations, or CAI. Does not include teaching of programming. (UW Fox Valley only)

Prerequisite: None.

CS 120 Introduction to E-Commerce (3 units)
This course will familiarize the student with the basics of e-commerce. Major topics include the basics of the internet, entrepreneurship, the creation of a business plan, financing, web site design, and e-business management. Students will develop a background in electronic commerce technology through exploring infrastructure and emerging technical issues in support of e-commerce. (UW Fox Valley only)

Prerequisite: None.

CS 125 Web Site Development (3 units)
This is an introductory course on website design and online branding. It covers elementary aspects of popular web development software packages. You will learn to create complete websites using responsive design and web services. The expectations for prerequisite knowledge are only basic computer skills.

This course does not apply toward the Computer Science major or minor. Not open to students who have completed CS 271.

Prerequisite: None.

CS 142 Introduction to Computer Science and Programming (3 units)
In this course, students will be introduced to the science of computing. The history of Computer Science, Elementary computer organization, big ideas in computing, algorithmic problem solving, and introductory programming concepts will be discussed. Programming concepts include data types, expressions, input/output, conditional statements, repetition, data processing, procedures and basic object-oriented principles. Programming concepts will be taught through the use of a specific programming language. The course assumes no previous programming experience. This course does not apply toward the Computer Science major or minor.

Prerequisite: Math 103 or PBIS 187 or PBIS 188 or PBIS 189 with a grade of C or better, or qualifying for either Math 104 or Math 171 via Math Placement Test. Not open to students who have completed CS 271.

CS 212 Discrete Structures (3 units)

This course focuses on discrete mathematical structures that are essential to computer scientists. In this course, students will develop their analytic and algorithmic thinking skills through practice with propositional and first-order predicate logic, various proof techniques, mathematical and structural induction, sets, functions, sequences, recurrence relations, algorithm analysis and computational complexity, the basics of counting, and an introduction to discrete probability.

Prerequisite: A grade of C or better in CS 262 AND (Math 171 or Math 206).

CS 216 Problem Solving and Programming Techniques in C++ (4 units)

Program design using both modular and object-oriented methods. Topics covered to include stream I/O, recursion, multi-dimensional arrays, sorting and searching, pointers and dynamic memory allocation, classes and abstract data types, and operator overloading. (UW Fox Valley only)

Prerequisite: Advanced high school mathematics, or previous programming experience, or consent of instructor.

CS 221 Object-Oriented Design and Programming I (3 units)

A first course in problem solving, software design, and computer programming using an object-oriented language. Problem solving/software design techniques include: flow charts, pseudo code, structure charts, structure charts, and UML class diagrams. Data structures and algorithms include: arrays, characters strings, linear search. Programming topics include; data types assignment statements, standard input/output, selection, repetition, functions/methods, parameters, scope of identifiers, debugging.

Prerequisite: A grade of C or better in Math 104 or Math 108 or Math 206 or CS 142, or qualifying for Math 171 via the Mathematics Placement Exam.

CS 247 Introduction to Usability (3 units)

This course presents the basic theory and professional views on design and usability, with an emphasis on human-computer interaction in web-based environments. The nature of life in general requires an understanding of how people think and act. What makes a well-designed door versus a poorly designed door? Understanding the answer scientifically is based on understanding people and usability, not the mechanics of doors. This class will focus on several basic principles that range from defining usability, design, sketches and how to evaluate usability. A case study will be conducted using high-level psychology and visual perception concepts that often pertain to usability.

Prerequisites: A grade of C or better in Comp Sci 142 or Comp Sci 221 or Info Sys 318.

CS 251 Computer Architecture and Assembly Language (3 units)

An introduction to RISC-based instruction set architecture. Topics include: data representation, assembly language programming, run-time storage management, pointers and references as exemplified in the C++ programming language, and introduction to system software.

Prerequisite: CS 221 with a grade of C or better.

CS 262 Object-Oriented Design and Programming II (4 units)

A second course in problem solving, software design, and computer programming using an object-oriented language. Problem solving/software design topics include: abstract data types, universal modeling language (UML), simple recursion, unit testing, event-handling, simple concurrency. Data structures and algorithms include: binary search, simple sorting algorithms, use of collection classes and their iteration protocols, sequential file processing. Additional topics include: inheritance, polymorphism, graphical user interfaces, simple use of threads.

Prerequisite: Mathematics 108 or equivalent with a grade of C or better, or qualifying for a higher level mathematics course via the Mathematics Placement exam, and CS 221 or equivalent with a grade of C or better.

CS 271 Data Structures (4 units)

A course surveying the fundamental methods of representing data and the algorithms that implement and use those data representation techniques. Data structures and algorithms include; linked lists, stacks, queues, trees, heaps, priority queues, hashing, searching, sorting, data compression, graphs, recursion. Analysis topics include: elementary big-O analysis, empirical measurements of performance, time/space trade-offs, and identifying differences among best, average, and worst case behaviors.

Prerequisite: CS 262 with a grade of C or better AND, for Computer Science majors, either Completion (with a grade of C or better) of or concurrent enrollment in CS 251.

CS 300 Artificial Intelligence (3 units)

This course is an introduction to the field of artificial intelligence. It is a survey of classical search in artificial intelligence and machine learning and an in-depth examination of a specific application area such as robotics, theorem proving, computer vision, natural language processing, etc. Students are expected to demonstrate mastery via computer programs using the techniques of artificial intelligence.

Prerequisite: CS 271 with a grade of C or better.

CS 310 Computer Organization and Design (3 units)

An introduction to digital logic and computer hardware organization. The students are introduced to elementary Boolean algebra and switching theory as related to computer architecture. Emphasis is given to the design of Central Processing Units, Arithmetic and Logic Units, and main memories. A comparison of alternate computer organizations is presented.

Prerequisite: A grade of C or better in CS 212 and CS 251.

CS 321 Algorithms (3 units)

Algorithm design techniques including brute-force, backtracking, divide-and-conquer, dynamic programming and greedy algorithms. Other topics include big-O and amortized analysis, recurrence relations in the analysis of recursive algorithms, numerical algorithms, pattern matching, data integrity, authentication, and encryption.

Prerequisite: A grade of C or better in CS 212 and CS 271.

CS 326 Computer Security (3 units)

This course is an introduction to computer security with an emphasis on software design principles and technical controls that help secure computer systems. After discussing foundational concepts in information security and assurance (e.g., the CIA triad, authentication, nonrepudiation, threats, attack vectors, risk assessment, security controls, plans, and policies), we will delve into the following topics: principles of secure software design and defensive programming, authorization and access control, and cryptography.

Prerequisites: A grade of C or better in CS 212 and CS 271.

CS 331 Programming Languages (3 units)

A study of programming languages. Topics covered include: formal syntactic description, methods of implementation, and language features such as recursion, block structure, string processing, and list processing. Specific high level programming languages are studied to demonstrate the use of these language features.

Prerequisite: CS 271 with a grade of C or better.

CS 341 Software Engineering I (3 units)

This course will provide an in-depth study and analysis of at least one large scale software system. Students will analyze, design, and partially implement an extensive software project. Case studies will address major system concerns such as specification, classification, inter-relationships, validation, and evaluation. Other topics include the use of UML, prototyping, data flow diagrams and CASE tools.

Prerequisite: CS 271 with a grade of C or better OR Junior-level standing and CS 262 with a grade of C or better.

CS 342 Software Engineering II (3 units)

Software Engineering is the second of a two-semester sequence on the topic of modern Software Engineering tools and techniques. Topics covered include Design Patterns, the Unified Modeling Language (UML), Component-based Software Development, Advanced OO Design and Analysis, Refactoring, and other techniques such as Extreme Programming. An extensive software development project will allow for practical application of the discussed techniques.

Prerequisite: CS 271 and CS 341, each with a grade of C or better. (Not currently offered)

CS 344 Mobile Application Development (3 units)

An introduction to the tools for developing mobile applications. Topics covered include: history of mobile development, using an appropriate IDE, emulating a mobile device, building a flexible user interface, understanding the application lifecycle, creating and managing multiple threads, creating and using web services and encrypting a completed project. A large mobile application will be created throughout the course.

Prerequisite: CS 262 with a grade of C or better.

CS 346 Web Software Development (3 units)

An introduction to the tools for developing internet applications. Topics covered include: Internet history, the HyperText Markup Language, graphic images and manipulation, multimedia, programming in the JavaScript and PERL languages.

Prerequisite: CS 262 with a grade of C or better.

CS 361 Database Systems (3 units)

An introduction to database processing with emphasis on database techniques, design, and modeling. Programming projects include implementation of selected database processing methods and the use of database software.

Prerequisite: A grade of C or better in CS 212 and CS 271

CS 371 Computer Graphics (3 units)

An introduction to the mathematics, data structures, and algorithms used to create graphical output in the programmable pipeline. Topics include graphics hardware, shaders, transformations in two- and three-dimensions, three-dimensional viewing, modeling three-dimensional shapes with polygon meshes, hierarchical modeling of three-dimensional objects, lighting and shading techniques, raster algorithms.

Prerequisite: CS 262 and Math-171 or Math 206 each with a grade of C or better.

CS 381 Theory of Computing (3 units)

An introduction to the basic concepts in the theory of computing. Topics covered will include automata theory, formal languages, Turing machines, the Chomsky Hierarchy, and undecidability.

Prerequisite: A grade of C or better in CS 271 AND (CS 212 OR Math 222).

CS 391 Data Communication and Computer Networks (3 units)

An introductory course which covers the basic concepts in data communication and computer networks. Topics covered will include the nature of data communication, characteristics of computer networks, the ISO-OSI network protocol layers, error detection and correction codes, and network performance considerations.

Prerequisite: A grade of C or better in CS 212, CS 251, and CS 271.

CS 421 Operating Systems (3 units)

An introduction to operating systems concepts. Topics covered include: interrupts, memory allocation, virtual memory techniques, process scheduling and synchronization, deadlocks, resource allocation, and file systems. A major programming project will be assigned to provide experience with operating system design.

Prerequisite: A grade of C or better in CS 251, CS 212 and CS 271.

NOTE: The Computer Science Department will accept courses similar to CS 421 from all other schools as counting toward the list of "elective courses," but not the list of "required courses."

CS 431 Compilers (3 units)

An introduction to compiler writing techniques for translating a higher level programming source language into a lower level target language. Topics to be covered include: definition of programming languages, lexical and syntactic analysis, low level code generation and optimization, run time systems, and error detection, reporting, and recovery. A major programming project will be assigned to provide experience with compiler design.

Prerequisite: A grade of C or better in CS 212 and a grade of C or better or concurrent registration in CS 331.

NOTE: The Computer Science Department will accept courses similar to CS 431 from all other schools as counting toward the list of "elective courses," but not the list of "required courses."

CS 474 Honors: Thesis (1-6 units)

Honors thesis projects include any advanced independent endeavor in the student's major field of study e.g., a written thesis, scientific experiment or research project, or creative arts exhibit or production. Proposals (attached to Independent Study contract) must show clear promise of honors level work and be approved by a faculty sponsor. Course title for transcript will be 'Honors Thesis.' Completed projects will be announced and presented to interested students and faculty.

Prerequisite: University Honors program and junior standing. Maximum of 6 units (crs.). A maximum of 3 units (crs.) can count as electives toward the Computer Science major (in all emphases) as long as at least one Computer Science faculty member is sponsoring the project.

CS 480 Topics in CS (3 units)

A topic of current interest in computer science will be investigated by faculty and qualified students. In addition to lectures by faculty, the students will be responsible for research and presentation of selected aspects of the topic. The course may be repeated for credit if the topic is different, and the student has the consent of department.

Prerequisite: Junior-level standing and consent of the instructor.

Practicum and Internship

The courses CS 399 Internship in Computer Science and CS 490 Practicum in Computer Science are designed to be the capstone courses for the major or minor. Either course satisfies the Writing Emphasis requirement in the Computer Science Department. Each semester a faculty member in the Computer Science department is assigned the duties of being the "Practicum & Internship Coordinator." The name of and the times to see the coordinator are posted in the departmental office each semester.

CS 399 Internship in Computer Science Credits (3 units)

An internship experience with a cooperating organization or corporation to provide on-the-job learning. This course satisfies the Writing Emphasis requirement for the Computer Science major.

The internship guidelines and application form are online.

Prerequisites: 75 credits toward graduation, at least six credits of 300-400 level CS courses, and completion of IS 208 (Professional Career Skills in Math and Natural Science, 1 credit) with a passing grade.

CS 490 Practicum in Computer Science (3 units)

A project-oriented course that brings together the material learned in previous computer science courses. The student will investigate and solve a problem(s) under supervision. This course satisfies The Writing Emphasis requirement for the Computer Science major.

The practicum guidelines and application form are online.

Prerequisites: 75 credits toward graduation, at least six credits of 300-400 level CS courses, and completion of IS 208 (Professional Career Skills in Math and Natural Science, 1 credit) with a passing grade.

Progress Checklist
Computer Science Major

Computer Science Major	
Computer Science Required:	Computer Science Electives:

Computer Science Required:

1. CS 212 _____
2. CS 221 _____
3. CS 251 _____
4. CS 262 _____
5. CS 271 _____
6. CS 321 _____
7. CS 331 _____
8. CS 341 _____
9. One of:
   1. CS 399 _____
   2. CS 490 _____

Computer Science Electives:

1. _____
2. _____
3. _____
4. _____

Mathematics Required:

1. Calculus _____
2. Statistics _____

Philosophy Required:

1. Phil 350 _____

Interdisciplinary Studies Required:

1. IS 208 _____

Progress Checklist
Computer Science Major - Systems and Theory Emphasis

Systems and Theory Emphasis	

Computer Science Required:
 	
1. CS 212 _____
2. CS 221 _____
3. CS 251 _____
4. CS 262 _____
5. CS 271 _____
6. CS 310 _____
7. CS 321 _____
8. CS 331 _____
9. CS 341 _____
10. CS 381 _____
11. CS 421 _____
12. CS 431 _____
13. CS 499 _____
14. One of:
   1. CS 399 _____
   2. CS 490 _____
	Computer Science Electives:
	
	1. _____
	2. _____
	
	Mathematics Required:
	
	1. Math-171 _____
	2. Math-172 or Math-256 _____
	3. Math-201 or Math-301 _____
	
	Philosophy Required:
	
	1. Phil 350 _____
	
	Interdisciplinary Studies Required:
	
	1. IS 208 _____

Progress Checklist
<u>Software Technology Major</u>

Software Technology Major	

Computer Science Required:
 	
  10. CS 221 _____
 	
  11. CS 247 _____
 	
  12. CS 262 _____
 	
  13. CS 271 _____
 	
  14. CS 341 _____
 	
  15. CS 344 _____
 	
  16. CS 346 _____
 	
  17. CS 399 _____
	INFO SYS Required:
	
	  5. INFO SYS 315 _____
	
	  6. INFO SYS 316 _____
	
	One of following INFO SYS Required:
	
	  3. INFO SYS 318 _____
	
	  4. INFO SYS 432 _____
	
	Philosophy Required:
	
	  2. Phil 350 _____
	
	Interdisciplinary Studies Required:
	
	  2. IS 208 _____

Student Notes