Computer Science

College of Engineering
and Computer Science

Department of Computer Science
Henderson C. Yeung, *Chair*
McKee Fisk Building, Room 207
(559) 278-4373
FAX: (559) 278-4197
e-mail: cs@cs.csufresno.edu
http://www.engr.csufresno.edu

B.S. in Computer Science
M.S. in Computer Science
Minor in Computer Science

The Department

Computer science is applied reasoning using both art and science: It requires the ability to communicate ideas through a combination of language and powerful technology. It is concerned with the interaction of humans and computers, as well as the application of computers to a myriad of specialized problems.

The goal of the Department of Computer Science is to offer programs to a diverse audience: (1) students interested primarily in computing, (2) students interested primarily in applying computing to some other field of study, and (3) students who wish to include computing as part of their general education.

Faculty and Facilities

The faculty comes from a variety of areas including computer systems and architecture, theoretical computer science, programming languages, software engineering, computer graphics, distributed systems and parallel processing, and neural networks. They have in common a desire to provide a program that will give the student a broad range of experience in computer science as well as the depth of education that will be needed in the student's later career, whether professional or academic.

Students and faculty have access to a networked environment of UNIX workstations (Sun Microsystems, Silicon Graphics, and Linux systems) and microcomputer laboratories of Macintosh and PCs. These systems are connected to campus and international networks.

The computer graphic and scientific visualization laboratory is equipped with Silicon graphics machines and the state-of-the-art Alias/Wavefront software.

Career Opportunities

Computer use pervades our society, and the industry supporting that use has grown rapidly. Graduates from this program find job opportunities in such diverse fields as computer design, software engineering, systems analysis, database design, computer graphics, and technical programming. Because of the strong theoretical foundation of the program, graduates are attractive to companies involved in computer manufacturing and to those industries using computers in high technology applications.

Our proximity to two of the largest computer use areas in the nation, Silicon Valley and Los Angeles, provides our graduates with a broad-based collection of potential employers. Graduates have also obtained exciting and challenging positions at Air Force and Naval bases in California. A significant proportion of our graduates pursue graduate studies. Students who obtain the master of science degree will be in an excellent position to pursue a Ph.D. degree.

Organizations

Student chapters of the Association for Computing Machinery (ACM) and the IEEE Computer Society are very active in the department. They organize field trips to major computer manufacturers and users in California. The ACM chapter sponsors the fall Programming Contest.

Computer science majors who have a distinguished academic record in computer science are invited to join Upsilon Pi Epsilon, the Honor Society for the Computing Sciences.

Co-op Program

Through the Cooperative Education program, students receive academic credit and are employed in computer-related industries. This is an excellent opportunity for a student to obtain experience, a reasonable salary, and college credit in this field.

Faculty

Henderson C. Yeung, *Chair*
Tarek Alameldin
Brent J. Auernheimer
Lan Jin
Walter Read
Shigeko Seki
Jerome Smith
Prudence Tovey
Grace C. N. Wei
J. Todd Wilson

Undergraduate Program

The bachelor's degree in computer science prepares students for careers in the computing industry or for graduate study. Combined with a minor in another field of study, the bachelor's degree allows students to utilize their computing expertise in a variety of specialized fields. The core and computer science theory courses are excellent preparation for students who intend to pursue an advanced degree in computer science.

For the computer science major, the department offers courses that represent both the core of study considered essential to all aspects of computing and advanced study sequences in particular fields of interest. The core classes introduce all majors to the spectrum of thought represented in computing. The advanced sequences allow the individual student to pursue concentrated work within such areas as computer architecture, artificial intelligence, databases, compilers, operating systems, computer science theory, computer graphics, software engineering, programming languages, networking distributed systems, and parallel processing. The department also offers topics courses to keep students informed of current advances and methods in computing.

In addition to courses designed for majors, the department offers courses intended to introduce computing to nonmajors. These courses will benefit any major who wishes to include computing in their undergraduate study.

Grade Requirements. All courses taken to fulfill major course requirements must be taken for a letter grade. All courses required as prerequisites for a course must be completed with a grade of *C* or better before registration will be permitted.

Administrative Academic Probation. A minimum Grade Point Average (GPA) of 2.0 must be maintained in all courses taken in the College of Engineering and Computer Science. Students who fail to maintain a 2.0 GPA in courses within their major may be placed on administrative academic probation. Failure to eliminate the grade point deficiency could result in disqualification from the College of Engineering and Computer Science.

Bachelor of Science Degree Requirements

Computer Science Major	*Units*
Major requirements	**60-64**
CSCI 40, 41, 60, 112, 113, 115, 117, 119, 144	(36)
Select seven courses from the following, including one sequence	(21-24)
CSCI 124, 126, 130, 134, 136, 146, 148, 150, 152, 154, 156, 164, 166, 172, 173, 174, 176, 177, 186, 188, 191T (max total 6 units)	
Approved Sequences:	
CSCI 124-126	
CSCI 134-136	
CSCI 144-146	
CSCI 144-148	
CSCI 150-152	
CSCI 156-ECE 146	
CSCI 164-166	
CSCI 172-173	
CSCI 176-177	
CSCI 186-188	
CSCI 198 or complete an additional second course in one of the sequences above	(3-4)
Additional requirements	**10**
MATH 75, 76; PHYS 2A and 2B or PHYS 4A, 4AL, 4B, 4BL	
General Education requirements	**51***
Electives and remaining degree requirements	**0-3**
Total	**124**

*Of the 51 required General Education units, 6 units will be satisfied by the following courses in additional requirements: MATH 75 and PHYS 2A or 4A.

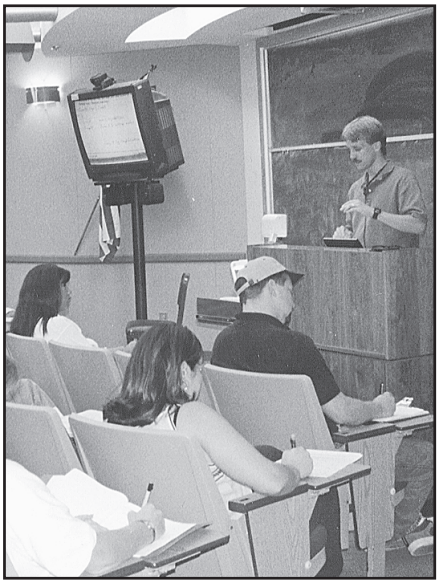
Note: Pass the Upper-Division Writing Exam or IE 182W (recommended to satisfy the upper-division writing skills graduation requirement).

Computer Science Minor

The Computer Science Minor requires 20 units of computer science courses consisting of CSCI 40, CSCI 41, and 12 units from CSCI 1, 60, 72, or upper-division courses. At least 6 of the 20 units must be upper division. No *CR/NC* courses will be accepted toward the Minor in Computer Science.

Suggested minor sequences (after completion of CSCI 40, 41):

- Artificial Intelligence: CSCI 60, 112, 117, 164, 166



- Computer Architecture: CSCI 112, 113, 176, 177
- Computer Graphics: CSCI 112, 172, 173
- Computer Languages: CSCI 60, 112, 115, 117, 134
- Database Emphasis: CSCI 60, 115, 124, 126, 144
- Scientific Computation: CSCI 60, 112, 154*, 172*
- Secondary Teaching: CSCI 60, 112, 113, 115, 117
- Software Engineering: CSCI 60, 112, 115, 150, 152
- System Software: CSCI 112, 113, 144, (146 or 148)
- Theory of Computation: CSCI 60, 119, 174, 186, 188

*CSCI 154 and 172 have a mathematics prerequisite. Note that these are only suggested combinations. While attention must be given to prerequisites, many combinations are available to interested students.

Note: The Computer Science Minor also requires a 2.0 GPA and 6 upper-division units in residence.

Graduate Program

The Master of Science degree program in Computer Science is designed to offer the advanced principles, applications, and current topics in computer science. Students who obtain the M.S. degree will be ready to do significant developmental work in the computer industry or in an important application area and will also be well qualified to pursue a Ph.D. degree.

Applicants may hold an acceptable bachelor's degree in any field of study and must submit Graduate Record Examination (GRE) scores.

To attain classified standing at the time of admission, an applicant must:

1. have a minimum grade point average of 2.75 in the last 60 units and
2. have completed the following undergraduate prerequisite courses or equivalents with a minimum grade point average of 3.0: CSCI 40, 41, 60, 112, 113, 115, 117, 119, 144, MATH 75, 76.

Applicants who do not meet the requirements 1 and 2 above may be admitted to conditionally classified standing to complete the remaining prerequisites at California State University, Fresno. Approved coursework up to a maximum of 10 units of the 30 units required for the M.S. degree can be taken concurrently with prerequisite courses by a student with conditionally classified standing.

To attain classified standing from conditionally classified standing, a student must complete the remaining prerequisite courses with a minimum grade point average of 3.0 and have earned a minimum grade point average of 3.0 in all coursework taken toward the M.S. degree in Computer Science.

(See also *Graduate Studies*)

Master of Science Degree Requirements

The Master of Science degree requires a minimum of 30 units after the completion of the baccalaureate degree according to the criteria below. At least 21 units of the total must be taken in 200-level courses in computer science. The undergraduate courses used toward the bachelor's degree or toward fully classified status may not be used toward the master's degree.

	Units
Required courses	**10**
CSCI 174 or 188, 200, 213, 217	
Electives	**9**
Three of the following: CSCI 226, 244, 246, 250, 252, 264, 272, 274, 282, 284	
Approved electives	**5-8**
Culminating experience	**3-6**
CSCI 298 or 299	
Total	**30**

A student must pass the Departmental Qualifying Examination prior to advancement to candidacy. One component of the exam will be used to satisfy the graduate writing skills requirement.

COURSES

Computer Science (CSCI)

1. Critical Thinking and Computer Science (3)
Prerequisite: intermediate algebra. Overview of the field of computer science with an emphasis on critical thinking skills. Problem-solving strategies, algorithm design, and data abstraction. Introduction to hardware, theoretical limitations of computers, and issues arising from the growing role of computers in society. G.E. Foundation A3.

5. Computer and Applications (3)
An introduction to the computer: tools, applications, and graphics. Overview of the components of computer systems; discussion on software systems, electronic mail, influence of computers on society and the future of computing; extensive hands-on experience with application tools and programming. PC (Windows) environment. *CR/NC* grading only. (2 lecture, 2 lab hours)

7. Computer Literacy (3)
Overview of the history of computing, a presentation of the components of computer hardware and software systems, a study of applications, programming, electronic mail, societal impact, and the future of computing. Macintosh environment. (2 lecture, 2 lab hours)

10. Introduction to BASIC Programming (1)
Prerequisite: elementary algebra. Introduction to structured programming techniques using the programming language BASIC. Topics include input/output, branching, looping, subroutines, and computer graphics. No prior experience required.

15. C and C++ Programming (2)
Prerequisite: programming experience in a major high-level language, e.g., BASIC, COBOL, FORTRAN, Pascal. An introduction to the C and C++ programming languages. Types, operators, expressions, flow of control, functions, pointers, and arrays. Standard libraries and programming tools. Emphasis on programming projects.

20. FORTRAN Programming (4)
Prerequisites: Students must take the ELM exam; students who do not pass the exam must record a grade of *C* or better in a college-taught intermediate algebra course; trigonometry. Introduction to programming in FORTRAN with emphasis on program design, debugging, and documentation. Elementary applications and structured programming for algorithm development. (3 lecture, 2 lab hours) (CAN CSCI 4)

40. Introduction to Programming and Problem Solving (4)
Prerequisites: Students must take the ELM exam; students who do not pass the exam must record a grade of *C* or better in a college-taught intermediate algebra course; trigonometry. Introduction to problem solving, algorithm development, procedural and data abstraction; program design, coding, debugging, testing, and documentation; a high-level programming language. (3 lecture, 2 lab hours)

41. Introduction to Data Structures (4)
Prerequisite: CSCI 40. Programming methodology, program correctness. Review of data types. Data structures: linear and nonlinear structures, files. Implementation of data structures. Recursion. Searching and sorting. (3 lecture, 2 lab hours)

60. Foundations of Computer Science (4)
Prerequisites: CSCI 40 (may be taken concurrently.) Abstraction, iteration, induction, recursion, complexity of programs, data models, and logic. (3 lecture, 2 lab hours)

72. Introduction to Computer Graphics (3)
Comprehensive overview of computer graphics. Geometry, color, hardware devices, surfaces and materials, lighting and shading, polygonal models, textures, fractals, rendering, animation, and production techniques. Case studies, examples, films, and video displays from actual systems.

101. Computational Foundations for Bioinformatics (3)
Prerequisite: C SCI 1, BIOSC 140A. Computational approaches to problems in molecular biology; Algorithms, heuristics, strings, and graphs. Sequence comparison and multiple alignment. Selected topics such as scripting, visual programming, laboratory workflow, databases, and queries. (2 lecture, 2 lab hours)

105T. Workshop on Computer Languages (1-3; max total 6)
Prerequisite: CSCI 40 or permission of instructor. Workshops in the use of various high-level programming languages or other selected languages in areas of database, statistical computation, or operating systems.

112. Introduction to Computer Systems (4)
Prerequisite: CSCI 41. Computer arithmetic. Von Neumann architecture. Instruction sets, data types, formats, addressing. Register and ALU organization. Memory hierarchy. I/O. Bus organization. Study of one or more assembly languages. Basics of implementation of higher-level languages. (3 lecture, 2 lab hours)

113. Introduction to Computer Organization (4)
Prerequisite: CSCI 41. Fundamental issues of computer design at register-transfer level. Logical design of basic combinational and sequential modules. Organization and design of major functional blocks: ALU, CPU, memory, cache, input/output, hard-wired and microprogrammed control. Simulation of computer organization. Introduction to high-performance superscalar computer organization. (3 lecture, 2 lab hours)

115. Algorithms and Data Structures (4)
Prerequisites: CSCI 41, 60; MATH 75. Review of basic data structures. Graph, search paths, and spanning trees. Algorithm design and analysis of sorting, merging, and searching. Memory management, hashing, dynamic storage allocation. Integration of data structures into system design. (3 lecture, 2 lab hours)

117. Structures of Programming Languages (4)
Prerequisites: CSCI 41, 60, and CSCI 119. General concepts and paradigms of programming languages; scope and binding rules, applications and implementations of language concepts. Languages selected from: ADA, ICON, Miranda, ML, MODULA 2, OCCAM 2, PROLOG, LISP, Scheme, Smalltalk. (3 lecture, 2 lab hours)

119. Introduction to Finite Automata (4)
Prerequisites: CSCI 41, 60. Strings, languages, and fundamental proof techniques. Regular expression, regular grammar, regular languages, finite automata, their interrelationship, and their properties. Introduction to context-free languages. (3 lecture, 2 lab hours)

124. Introduction to File Processing (3)
Prerequisite: CSCI 115. Definition of file components, access methods, and file operations. Algorithms for efficient implementation of data structures; characteristics of bulk storage media for mainframe

and microcomputers. Introduction to database management systems.

126. Database Systems (3)
Prerequisite: CSCI 124. Database concepts; hierarchical and relational network models; object-oriented data models. Data normalization, data description languages, data manipulation languages, and query design.

130. Web Programming (3)
Prerequisite: CSCI 115. Programming for the World Wide Web. Web servers and clients, Internet and Web protocols, and mark-up languages. Client-side scripting, including both gateway and filter-based approaches. (2 lecture, 3 lab hours) (Formerly CSCI 191T)

134. Compiler Design (3)
Prerequisites: CSCI 112, 115, 119. Syntax and semantics of programming languages. Lexical analysis, parsing techniques, parser generator, SLR and LALR parsing. Introduction to symbol table organization and semantic routines. Compiler generators.

136. Compiler Construction (3)
Prerequisite: CSCI 134. Advanced topics in compiler design. Type checking. Run-time storage management. Intermediate code generation. Interpreters. Error recovery techniques. Code generation and optimization.

144. Introduction to Operating Systems (4)
Prerequisites: CSCI 41 and CSCI 112 or ECE 118. Operating system history and services. File systems. Memory management. Process management — concurrent processes, communication, semaphores, monitors, deadlocks. Resource management — processor and disk scheduling. Security and protection mechanisms. (3 lecture, 2 lab hours)

146. Systems Architecture (3)
Prerequisites: CSCI 113, 144. An in-depth analysis of one or more operating systems — system data structures, hardware architecture, shell and kernel functions, I/O routines, interrupt handling. Other topics may include parallel hardware architectures, performance analysis.

148. Systems Programming (3)
Prerequisites: CSCI 113, 144. Topics include implementation of operating system components and modification of existing systems. Device drivers, memory management, communication networks, and file systems will be examined. Projects will be emphasized.

150. Introduction to Software Engineering (3)
Prerequisite: CSCI 41. History, goals, and motivation of software engineering. Study and use of software engineering methods. Requirements, specification, design, implementation, testing, verification, and maintenance of software systems. Team programming. (2 lecture, 3 lab hours)

152. Software Engineering (4)
Prerequisite: CSCI 150. In-depth examination of techniques for specification, design, implementation, testing, and verification of software. Human-computer interfaces. Formal methods of software development. Use of software engineering tools for the development of substantial software projects. (3 lecture, 3 lab hours)

154. Simulation (3)
Prerequisites: CSCI 41, 60; MATH 75. Simulation as a tool for the study of complex systems in computer science, statistics and operations research. Generating random variables. Review of principles behind and examples of simulation languages.

156. Internetworking Systems and Protocols (3)
Prerequisite: CSCI 144 or permission of instructor. Review of underlying network technologies. Application-level interconnections, network architectures, addressing, mapping abstract addresses to physical addresses, routing datagrams, error and control messages, protocol layering, gateways, subnets. Client-server interactions. Upper layers of protocol stacks. (2 lecture, 3 lab hours)

164. Artificial Intelligence Programming (3)
Prerequisite: CSCI 117. Introduction to problem-solving methods from artificial intelligence. Production systems. Knowledge-based systems. Machine learning. Topics chosen from fuzzy logic, neural network models, genetic algorithms. Verification, validation, testing.

166. Principles of Artificial Intelligence (3)
Prerequisite: CSCI 164. Analysis of knowledge-based and neural models, including self-organization, sequential learning models, neurally inspired models of reasoning and perception. Integration of different paradigms.

172. Computer Graphics (4)
Prerequisites: MATH 76, CSCI 41, and (CSCI 112 or ECE 118). Hardware devices, raster graphics, device in dependence, graphic data structure and representations, interactive techniques, and algorithms for the display of two- and three-dimensional objects, graphic transformations, graphics standards, modeling, animation, VRML, and scientific visualization. (3 lecture, 2 lab hours)

173. Advanced Computer Graphics (4)
Prerequisite: CSCI 172. Visible surface algorithms, lighting and shading, textures, curves and surfaces, computer-aided design, advanced modeling techniques, solid modeling, advanced raster graphics architecture, advanced geometric and raster algorithms, user interface, ray tracing, animation techniques, and fractals. (3 lecture, 3 lab hours)

174. Design and Analysis of Algorithms (3)
Prerequisites: CSCI 115, 119. Models of computation and measures of complexity, algorithms for sorting and searching, set representation and manipulation, branch and bound, integer and polynomial arithmetic, pattern-matching algorithms, parsing algorithms, graph algorithms, NP-complete problems.

176. Parallel Processing (3)
Prerequisites: CSCI 113, 144. Characteristics, and classification of computer systems. Notion and realization of parallelism. Pipeline design techniques. Vector processing. Array processing. Multiprocessing. Multiprocessing vs. multicomputers. Shared memory vs. message-passing, problem solving, and parallel programming. Architectural trends.

177. Distributed Computer Systems (3)
Prerequisites: CSCI 113, 144. Characteristics and design of distributed systems. Application and network interconnectivity. Enterprise computing. Distributed data and transaction management. Distributed operating systems. Distributed problem solving and programming.

186. Formal Languages and Automata (3)
Prerequisite: CSCI 119. Introduction to formal language theory. Context-free grammars, context-sensitive grammars, unrestricted grammars, graph grammars, and rewriting systems; properties of context-free languages, push-down automata.

188. Introduction to Computability (3)
Prerequisite: CSCI 119. Introduction to computability and complexity. Turing machines, recursive functions, reduction, undecidability, classes P and NP, and intractable problems.

190. Independent Study (1-3; max total 6)
See *Academic Placement — Independent Study*. Approved for ***RP*** grading.

191T. Proseminar (1-3; max total 15)
Prerequisite: permission of instructor. Presentation of selected topics in computer science.

194. Cooperative Education (1-4; max total 8)
Prerequisites: courses appropriate to the work experience; approval by major department cooperative education coordinator. Integration of work experience with academic program, individually planned through coordinator. ***CR/NC*** grading only.

198. Project (3)
Prerequisite: senior standing in computer science or permission of instructor and approved subject. See *Criteria for Thesis and Project*. Study of a problem under the supervision of a faculty member. Presentation by the student in a seminar setting and a final report are required. Satisfies the senior major requirement for the B.S. in Computer Science. Approved for ***RP*** grading.

GRADUATE COURSES

(See *Course Numbering System.*)

Computer Science (CSCI)

200. Introduction to Research in Computer Science (1)
Prerequisite: classified standing in computer science. Orientation to the graduate program, introduction to research methodology, and discussion of possible project and thesis topics.

213. Computer Organization (3)
Prerequisites: CSCI 112 and 113 or permission of instructor. Organization of memory, I/O, and processors. Computer busses. Microprogramming and instruction execution. Interrupts. Data communications.

217. Programming Language Principles (3)
Prerequisite: CSCI 117 or permission of instructor. Advanced topics in programming languages: concurrency, exceptions, types, procedures, execution models. Introduction to the formal specification of programming languages: syntax specification, semantic specification.

226. Advanced Database Systems (3)
Prerequisites: CSCI 126 and 144. Implementation of database systems on modern hardware systems. Operating system design issues, including buffering, page size, prefetching, etc. Query processing algorithms; design of crash recovery and concurrency control systems. Implementation of distributed databases and database machines.

230. Advanced Web Application Development (3)
Prerequisite: CSCI 130 or permission of instructor. Application development for the World Wide Web. Three-tier architecture; authentication, capability, and session management; versioning and open-source development. Case studies and project work. (Formerly CSCI 291T)

244. Operating Systems (3)
Prerequisite: CSCI 144. Operating system functions. Performance monitoring and fine-tuning. Network operating system design. Concurrency, analysis of deadlock. Selected topics from current research.

246. Computer Architecture (3)
Prerequisite: CSCI 144. Examination and comparison of RISC and CISC architectures. Parallel processors, multiprocessors, dataflow machines. Database machines. Selected topics from current research.

250. Advanced Software Engineering (3)
Prerequisite: CSCI 150 or permission of instructor. Theoretical and practical aspects of software engineering emphasizing requirements analysis, specification, design, coding, testing, correctness, maintenance, and management. Examination of reliability, performance, and software metrics.

252. Software Development Environments (3)
Prerequisite: CSCI 150. Overview of modern software engineering environments including structured editors, programmer's assistants, and tools for software cost estimation, testing, scheduling, specification, and verification. Relationship between artificial intelligence and software engineering.

264. Artificial Intelligence (3)
Prerequisite: CSCI 164 or ability to program in Lisp and Prolog. Software technology for artificial intelligence systems, including expert systems. Knowledge-based and rule-based systems. Explanation and learning. User-oriented interfaces.

272. Computer Graphics (3)
Prerequisite: CSCI 172 or permission of instructor. 3-D transformations, visible-surface algorithms, shading, textures, curves and surfaces, computer-aided design, advanced modeling techniques, solid modeling, advanced raster graphics architecture, advanced geometric and raster algorithms, user interface, ray tracing, animation techniques, and fractals.

274. Combinatorial Algorithms (3)
Prerequisite: CSCI 174. Design and analysis of efficient algorithms for combinatorial problems. Network flow theory, matching theory, augmenting-path algorithms, branch-and-bound algorithms, data structure techniques for efficient implementation of combinatorial algorithms, analysis of data structures, application of data structural techniques to sorting, searching, and geometric problems.

282. Theory of Computation (3)
Prerequisite: CSCI 188 or permission of instructor. General models of computation, recursive functions, undecidable problems, propositional calculus, predicate calculus, complexity classes, NP-complete problems.

284. Automata Theory (3)
Prerequisite: CSCI 186 or permission of instructor. Formal languages, abstract machines, algebraic approach to automata, term rewriting systems, formal power series, cryptography, parallel computation.

290. Independent Study (1-3; max total 6)
Prerequisite: approval of department. See *Academic Placement — Independent Study*. Approved for ***RP*** grading.

291T. Seminar (1-3; max total 9)
Prerequisite: approval of instructor. Special topics in computer science of current interest and importance.

298. Research Project (3)
Prerequisite: advancement to candidacy. See *Criteria for Thesis and Project*. Independent investigation of an advanced topic as the culminating requirement for the master's degree. Approved for ***RP*** grading.

299. Master's Thesis (3-6; max total 6)
Prerequisite: advancement to candidacy. See *Criteria for Thesis and Project*. Preparation, completion, and submission of an acceptable thesis for the master's degree. Approved for ***RP*** grading.

IN-SERVICE COURSE

(See *Course Numbering System.*)

Computer Science (CSCI)

391T. Topics in Computer Science (1-6; repeatable for credit with different topics)