COMPUTER SCIENCE

Undergraduate and Graduate Program

CATALOG EXTRACT 2001-2002

For more information: http://www.csun.edu/~sgs/compsci/

COMPUTER SCIENCE

Chair: Steven Stepanek

College of Engineering and Computer Science

Department Office: Engineering 2403
Phone: (818) 677-3398
Office Manager: Sally Sawchuk

FACULTY: Jack Alanen, Prasanta Barkataki, Michael Barnes, Richard Covington, Steven Fitzgerald, Peter Gabrovsky, Philip Gilbert, Robert Henderson, Jaime Kim, Robert Lingard, Richard Lorentz, Robert McIlhenny, Gloria Melara, John Motil, John Noga, Son Pham, David Salomon, Diane Schwartz, Peter Smith, Steven Stepanek (Chair), and Brenda Timmerman.

EMERITUS FACULTY: Philip Gilbert, Ruth Horgan, Larry Lichten, Dorothy Miller, John Motil.

GRADUATE COORDINATOR: Richard Lorentz.

THE MAJOR: Computers have become crucial in almost every field from accounting to zoology. In every aspect of science, art, business and entertainment, computer applications abound. For students who think logically, enjoy solving problems, and are persistent, Computer Science is a good study choice.

Our department's pragmatic approach to computer science offers hands-on design experience as well as theoretical knowledge. That's an advantage on the job because graduates have experience in design, implementation, and testing.

We teach logical thinking and creative problem-solving skills and we emphasize communication, both written and spoken. Good communication is vital in working with clients, and helping them understand how the software works. Classes often incorporate a team approach, requiring clear communication among members as they solve a problem and explain their solution to others.

Classes are small, averaging 25 students. They are taught by faculty members, not graduate students. The faculty works on such fascinating topics as virtual reality, high-speed networks, parallel computing, and Internet technology. Students work alongside them in the department labs, most of which have been recently equipped with state-of-the-art computers.

Students gain extra experience in the ACM Student Club (part of a national organization), which runs Saturday workshops and social activities. A strong scholarship program is available.

The Computer Science Program is accredited by the Computer Science Accreditation Commission of the Computing Science Accreditation Board.

CAREERS: A computer science major can have a career as a software engineer, designing, implementing, testing and maintaining large software systems. Careers are available in such specialties as computer graphics, computer security, robotics, expert systems, distributed systems and networking. Many companies employ computer science majors as computer systems and networking. Many companies employ computer science majors as computer systems administrators. The degree can lead to a career in almost any industry, including aerospace, manufacturing, banking, health and education.

PROGRAMS OFFERED

The undergraduate program, leading to a B.S. in computer science, provides a broad knowledge of computing. It consists of core courses in programming languages, computer system organization and operating systems, data structures, computation theory, computer logic, and societal implications in computing.

The freshman year program includes courses in mathematical analysis, algorithms and programming, computer architecture and assembly language, symbolic logic, and data structures and program design. Sophomores take courses in linear algebra, computer organization, programming language concepts, files and data bases, and fundamental mathematical concepts.

As juniors, students take courses in digital systems; combinatorial algorithms; automata, languages, and computation; operating systems and system architecture; and program design techniques, along with a course in either probability or statistics.

With the help of an advisor, seniors formulate a Concentrated Studies Package related to their career objectives. They also take a course in societal issues in computing and a course in numerical analysis or methods.

Students who hold a recognized bachelor's degree from an accredited institution may apply for the Second Bachelor's degree program.

A minor in computer science calls for 32 units of study, including courses in computer architecture and assembly language, algorithms and programming, data structure and program design, computer organization, programming language concepts, and files and data bases, along with a choice of electives.

Students in the M.S. program complete 30 units of graduate work, including a 6 units thesis or graduate project.

The core of the graduate program comprises advanced course in computation theory, algorithms and data structures, system architecture, and software engineering. The electives may be chosen to either form a concentration in an area of specialization or to provide a broadly based program of study, whichever is more consistent with the selected thesis or graduate project.

ACADEMIC ADVISEMENT

Each student meets with a faculty advisor during the Sophomore year and plans a concentrated studies program appropriate to the student's personal career objectives. For information regarding

the assignment of an advisor, please contact the Department Office.

Prior to the formation of their Graduate Committees, graduate students are advised by the Graduate Coordinator. After the formation of their Graduate Committees, graduate students are advised by the Committee Chair. All areas of specialization, courses and COMP 698 must be approved in advance by the Committee Chair.

ENGINEERING/COMPUTER SCIENCE SCHOLARSHIP INFORMATION

The College of Engineering and Computer Science administers a substantial undergraduate scholarship program, disbursing approximately $40,000 each year to high-achieving engineering and computer science students. The College also administers memorial scholarships and scholarships donated by friends of the University.

Applications and information are available in mid-February with applications due in mid-April. Specific dates and further information can be obtained from the Office of the Dean.

REQUIREMENTS FOR THE BACHELOR OF SCIENCE DEGREE (starting Fall 2001)

The B.S. in Computer Science program requires a total of 131 units, including general education requirements, pre-major core, major core, and 15 unit sequence of approved elective courses referred to below as a Concentrated Studies Package. A Computer Science major must complete a minimum of 18 residency units of upper division computer science courses which include 12 units in the Concentrated Studies Package, in addition to all other institutional residency requirements.

Requirement for Admission to the Major

To qualify for admission into the Computer Science major program, students must first complete a pre-major program in Computer Science consisting of eight lower division courses covering math, computer science and the university General Education requirements for Basic Subjects (Section A).

Grade requirements for admission into the Computer Science major program:

1. Overall G.P.A. of 2.0 in all courses taken to meet the requirements of the Computer Science pre-major program and G.E. Basic Subjects

2. Overall G.P.A. of 2.0 in all courses taken at CSUN

3. Grade of "C" or better in MATH 150A

After successfully completing all requirements for the pre-major in Computer Science (including G.E. Basic Subjects), students may apply for admission into the Computer Science major program by submitting to Admissions and Records a Change of Major Form signed by the Computer Science Department Chair or designee. Admission into the Computer Science major program is required prior to enrolling in upper division Computer Science courses.

Special Grade Requirements

Note: No grade lower than a "C" will be accepted on transfer from another institution to satisfy Computer Science requirements. No CSUN grade lower than a "C-" will be accepted for courses required in the Computer Science program.

LOWER DIVISION REQUIRED COURSES (37 UNITS)

The following eight lower division courses constitute the **Computer Science Pre-Major** program:

General Education Basic Subjects:
A.1: Written Composition
A.4: Oral Communication

(Subsections A.2 and A.3 are satisfied by the requirements of the Computer Science program.)

COMP	110/110L	Introduction to Algorithms and Programming 4
COMP	122	Computer Architecture and Assembly Language 3
COMP	182/182L	Data Structures and Program Design 4
MATH	150A	Mathematical Analysis I 5
MATH	150B	Mathematical Analysis II 5
PHIL	230	Symbolic Logic I 4

The following four lower division courses are part of the requirements of the **Computer Science Major** program. Prior to enrolling in these courses, students must complete all of the Computer Science Pre-Major requirements listed above. Computer Science Pre-Major students may not enroll in these courses without the consent of the instructor.

COMP	222	Computer Organization 3
COMP	232	Concepts of Programming Languages 3
COMP	282	Advanced Data Structures 3
MATH	262	Introduction to Linear Algebra 3

LOWER DIVISION ELECTIVES (12-15 UNITS)

Select one of the following science sequences (8-10 Units):

Physics	220A, 220AL, 220B, 220BL
Biology	106, 106L, 107, 107L
Chemistry	101, 101L, 102, 102L

Note: BIOL 107/107L has a prerequisite of CHEM 101.

Select an additional science course with corresponding lab outside of the sequence selected above (4-5 Units):

Biology	106, 106L
Chemistry	101, 101L
Geography	101, 102 (Lab)
Geography	103, 105 (Lab)
Geology	101, 102 (Lab)
Geology	110, 112 (Lab)
Physics	220A, 220AL

UPPER DIVISION REQUIRED COURSES (23 UNITS)

Before taking upper division courses in Computer Science, students must be admitted to the Computer Science major / minor programs or the Computer Engineering major program.

Note: All students must attempt the upper division writing proficiency examination prior to enrolling in any 400 level Computer Science course. The upper division writing proficiency examination must be passed prior to enrolling in COMP 450.

COMP	310	Automata, Languages & Computation	3
COMP	322/322L	Introduction to Operating Systems and System Architecture	4
COMP	380	Introduction to Software Engineering	3
MATH	326	Fundamental Concepts of Mathematics	3
EE	320/320L	Theory of Digital Systems and Lab	4
COMP	450	Societal Issues in Computing	3
MATH	482	Combinatorial Algorithms	3

UPPER DIVISION ELECTIVES (20-21 Units)

Select one of the following (3 units):

MATH	340	Introductory Probability	3
MATH	441	Introduction to Statistical Inference	3

Select one of the following (2-3 Units):

COMP	370	Numerical Methods for Computing	3
ME	309	Numerical Analysis for Engineering Systems	2
MATH	481A	Numerical Analysis	3

Concentrated Studies Package (15 Units)

CONCENTRATED STUDIES PACKAGE

Note: a grade of "C-" or better is required in all courses in the concentrated studies package.

All Computer Science majors are required to select a set of elective classes (Concentrated Studies Package) as part of their upper division program of study. A Concentrated Studies Package consists of a minimum of 15 units of upper division classes in Computer Science. All 400 and 500 level Computer Science courses (with the exception of COMP 450, 494 and 499) are pre-approved for inclusion in a Concentrated Studies Package.

A student should discuss their career goals with a Computer Science advisor prior to attempting any course work in their Concentrated Studies Package. The advisor will suggest appropriate sequences of Concentrated Studies Package courses for the student to select from.

Requests for inclusion of an upper division course within a Concentrated Studies Package that does not meet the requirements stated above must be approved by the student's advisor and the department chair prior to enrollment in the course.

TOTAL UNITS IN THE PRE-MAJOR AND MAJOR (92 Units)

GENERAL EDUCATION (33 Units)

Subsection A.2, Subsection A.3, Section B and Section E are considered satisfied by courses in this major.

TITLE 5 (6 Units)

In addition to General Education requirements, all students must fulfill Title V requirements in institutions and ideals, the Constitution of the United States, and state and local government.

TOTAL UNITS REQUIRED FOR DEGREE: 131

MINOR IN COMPUTER SCIENCE

COMP	110/110L	Introduction to Algorithms and Programming	4
COMP	122	Computer Architecture and Assembly Language	3
COMP	182/182L	Data Structures and Program Design	4
COMP	222	Computer Organization	3
COMP	232	Concepts of Programming Languages	3
COMP	282	Advanced Data Structures	3

Select one course from the following list (3 Units):

COMP	310	Automata, Languages and Computation	3
COMP	322/322L	Introduction to Operating Systems & System Architecture	4
COMP	380	Introduction to Software Engineering	3

Note: COMP 310 has a prerequisite of PHIL 230 and MATH 326

UPPER DIVISION COMPUTER-RELATED COURSES (9 Units)

Needs to be approved by the Chair.

TOTAL UNITS IN THE MINOR (31 Units)

REQUIREMENTS FOR THE MASTER OF SCIENCE DEGREE

Students in the M.S. program complete 30 units of graduate work, including a 6 unit thesis or graduate project.

Requirements for Admission

For admission to the Master of Science program, applicants must meet the requirements of the University as listed in the catalog, take the Graduate Record Examination (General Test), submit the results to the University, and be accepted to the program by the Computer Science Department. Each applicant's transcripts and GRE scores will be reviewed by the Computer Science Department to determine if the student shows high promise of success in the program.

To attain fully classified graduate status in the program, students must complete any required prerequisite undergraduate material, pass the Upper Division Writing Proficiency Exam and have a 3.0 grade point average for all work taken as a Conditionally Classified Student. Information about the prerequisite material can be obtained from the graduate coordinator.

Special Requirements

Each M.S. candidate must submit a proposal for a thesis or graduate project to be done under the supervision of a faculty member. When the thesis or project is approved by that faculty member, the graduate coordinator and the department, the proposal becomes a contract between the student and the department as to the work to be done for the thesis or graduate project. A three member project/thesis committee is formed with that faculty member as its chair. When the work is done, the student must prepare a report and defend or present results of the thesis or graduate project before the committee. The report and presentation must be approved by the student's project/thesis committee.

No course taken more than seven years prior to the date on which all requirements for the degree are completed may be counted as part of the 30 units in the degree program. No time limit applies to courses taken to satisfy M.S. prerequisite requirements.

REQUIRED COURSES (15 Units)

Breadth Requirement (9 Units)

The student will select one course from each of three areas of study. The areas of study and the courses available for selection in each area are shown below:

ALGORITHMS:
COMP 610 Data Structures and Algorithms 3

SYSTEMS:
COMP 620 Computer System Architecture 3

SOFTWARE ENGINEERING:
COMP 680 Software Engineering 3

FOUNDATIONS:
COMP 615 Advanced Topics in Computation Theory 3
COMP 630 Formal Semantics of Programming Languages 3

Project/Thesis (6 Units)

COMP 696 Directed Graduate Research 3
COMP 698 Thesis or Graduate Project 3

ELECTIVES (15 Units)

Courses at the 400, 500 and/or 600 level in a single area of specialization approved by the Graduate Coordinator, Department Chair, and Project/Thesis Committee Chair (6 units must be at the 500 level or above).

TOTAL UNITS REQUIRED FOR DEGREE: 30

COMPUTER SCIENCE COURSE LIST

LOWER DIVISION

100. Computers: Their Impact and Use (3)
An introduction to the uses, concepts, techniques, and terminology of computing. Places the possibilities and problems of computer use in historical, economic, and social contexts. Shows how computers can assist in a wide range of personal, commercial, and organizational activities. Typical computer applications, including word processing, spreadsheets, and databases. Not open to Computer Science majors. (Available for General Education, Applied Arts and Sciences.)

101. Introduction to Algorithms (2)
An introduction to the design, development and expression of algorithms. Algorithms and their stepwise refinement. Expression of algorithms in a formal language. This course is intended to be a first course in a two-course sequence, the second being a programming language laboratory (see COMP 105). Not open to students who have completed COMP 110/110L.

105. Computer Programming (1)
Prerequisite: COMP 101 or 110/110L or 106. Instruction and practice in a particular computer programming language as listed below. One three hour laboratory per week per unit.

COMP	105BAS	Computer Programming in BASIC . .1
COMP	105C	Computer Programming in C 1
COMP	105COB	Computer Programming in COBOL . 1
COMP	105FOR	Computer Programming in FORTRAN 1
COMP	105MOD	Computer Programming in MODULA 1
COMP	105PAS	Computer Programming in PASCAL 1

106. Computing in Engineering and Science (2)
Corequisites: MATH 150B. An introduction to computing, problem solving and programming intended for science and engineering majors. Programming practice in a high level structured language. Lab projects involve both micro computers and main frames.

106L. Computing in Engineering and Science Laboratory (1)
Corequisite: COMP 106. One three-hour laboratory per week.

108. Orientation to Computer Science (3)
Corequisite: Math 105 or Math 150A. An introduction to the Computer Science major and profession. The main focus is on developing problem solving, algorithm development, and programming skills, and acquiring critical thinking abilities especially when applied to Computer Science. There is additional emphasis on orientation to the University, campus resources, study skills, motivation, and career awareness. Recommended for incoming Computer Science majors with limited computing experience as well as those considering a major in Computer Science. Not a required course in the major.

110. Introduction to Algorithms and Programming (3)
Corequisite: MATH 150A, Fresh Comp. Introduction to algorithms, their representation, design, structuring, analysis and optimization. Implementation of algorithms as structured programs in a high level language.

110L. Introduction to Algorithms and Programming Laboratory (1)
Corequisite: COMP 110. One three-hour laboratory per week.

122. Introduction to Computer Systems (3)
Prerequisite: COMP 110, 110L, Fresh Comp. Corequisite: MATH 150B. An introduction to computer architecture, assembly language programming, system software and computer applications. Number systems and data representation. Internal organization of a computer. Primitive instructions and operations. Assembly language. Language translation principles. Overview of operating systems.

160. Navigating the Internet (1)
Prerequisite: Prior usage of a computer equivalent to COMP 100. Corequisite: COMP 160L. The Internet and its role as the backbone of the information superhighway is changing the way a growing percentage of the public communicates, gathers information for research and persona knowledge, and relaxes. Methods of accessing computer-based information on the Internet will be explored and the social implications of the existence of the information highway will be studied. Topics covered include: Internet, World Wide Web, HTML and homepages, commercialization of the Internet, and the Internet's role in society. (Available for General Education, Applied Arts and Sciences.)

160L. Navigating the Internet Lab (1)
Prerequisite: Prior usage of a computer equivalent to COMP 100. Corequisite: COMP 160. The lab is intended to reinforce the material presented in the lecture. Hands-on experience will be provided on the usage of various Internet services. A one unit lecture course accompanies the lab. (Available for General Education, Applied Arts and Sciences.)

165. Computer Graphics (2)
Corequisite: COMP 100 suggested. An introduction to computer graphics and elementary computer graphics programming. Survey of graphic devices and systems; programming languages and program procedures for graphic output; existing collections of program graphics procedures for inclusion in student programs.

165L. Computer Graphics Laboratory (1)
Corequisite: Comp 165. One three-hour laboratory per week.

182. Data Structures and Program Design (3)
Prerequisite: COMP 110, 110L, Fresh Comp. Corequisite: MATH 150B. Introduction to data structures and the algorithms that use them. Review of composite data types such as arrays, records, strings, and sets. The role of the abstract data type in program design. Definition, implementation, and application of data structures such as stacks, queues, linked lists, trees, and graphs. Recursion. Use of time complexity expressions in evaluating algorithms. Comparative study of sorting and searching algorithms.

182L. Data Structures and Program Design Laboratory (1)
Corequisite: COMP 182. One three-hour laboratory per week.

196. Experimental Courses in Computer Science (1-4)

222. Computer Organization (3)
Prerequisites: COMP 122, 182, 182L. Extension of basic addressing concepts to more advanced addressability such as base register and self-relative addressing. Comparative computer architecture focusing on such organizations as multiple register processors and stack machines. Basics of virtual memory input-output. Introduction to the concept of microprogrammable systems. Low level language translation process associated with assemblers. System functions such as relocatable loading and memory management. Application of data structure and hashing techniques to the above. Other related topics.

232. Concepts of Programming Languages (3)
Prerequisites: COMP 122, 182, 182L. Discussion of issues in the design, implementation, and use of high-level programming languages. Historical background. How languages reflect different design philosophies and user requirements. Technical issues in the design of major imperative (procedural) programming languages. Other approaches to programming: functional programming, logic programming, and object-oriented programming.

265. Graphics Programming (1)
Prerequisite: COMP 232 or COMP 165. Fundamentals of computer graphics (devices, languages, procedures and packages). Programming from simple graphics to a complete graphics package for a given application. Individual and team projects encouraged.

265L. Graphics Programming Laboratory (1)
Corequisite: COMP 265. One three hour laboratory per week.

282. Advanced Data Structures (3)
Prerequisite: COMP182, 182L, MATH 150B. An introduction to advanced data structures (particularly persistent structures) using object-oriented design. Main memory structures: hash tables, trees. Architectural foundations for files. Large-scale sorting. Hash-based persistent structures. Indexed files. Introduction to databases.

296. Experimental Courses in Computer Science (1-4)

UPPER DIVISION

310. Automata, Languages, and Computation (3)
Prerequisites: MATH 326, PHIL 230. Study of the relation of languages (i.e. sets of strings) and machines for processing these languages, with emphasis on classes of languages and corresponding classes of machines. Phrase structure languages and grammar. Types of grammars and classes of languages. Regular languages and finite state automata. Context-free languages and pushdown automata. Unrestricted languages and Turing Machines. Computability models of Turing, Church, Markov, and McCarthy. Applications to programming languages, compiler design, and program design and testing.

322. Introduction to Operating Systems and System Architecture (3)
Prerequisites: COMP 222, 282, and 105C or knowledge of C. Examination of the principal types of systems including batch, multi-programming, and time-sharing. Networked systems are also discussed. The salient problems associated with implementing systems are considered including interrupt or event driven systems, multi-tasking, storage and data base management, and input-output. Emphasis will be placed on some of the simple algorithms used to solve common problems encountered such as deadlocks, queue service, and multiple access to data. Projects will be implemented to reinforce the lectures.

322L. Operating Systems Laboratory (1)
Corequisite: COMP 322. One three hour laboratory per week.

332. Programming Language Semantics (3)
Prerequisites: COMP 222, 232, 282, 310. Study of the meaning of programming languages. Loop invariants, induction, and termination. Axiomatic systems, operational semantics, syntax directed translation, introduction to denotational semantics.

370. Numerical Methods for Computing (3)
Prerequisites: COMP 110, 110L, 122 and MATH 262. Error growth, linear and nonlinear systems, eigenvalues, interpolation, approximation, integration, ordinary and partial differential equations. Topics from parallel algorithms, optimization, high precision arithmetic, cryptography. Programming assignments will involve the comparison of algorithms and the analysis or error.

380. Introduction to Software Engineering (3)
Prerequisites: COMP 282, and COMP 222, or 232. Concepts and techniques for systems engineering, requirements analysis, design, implementation and testing of large scale computer systems. Principles of software engineering for production of reliable, maintainable and portable software products. Emphasis on functional analysis and structured design techniques. Topics include unit, integration and systems testing, configuration management, and software quality assurance practices. Participation in group activities involving analysis, design and implementation of a software intensive system. Introduction to Computer Aided Software Engineering (CASE)

396. Experimental Courses in Computer Science (1-4)

409. Fundamentals of Computer-Aided Manufacturing (2)
Prerequisites: Senior standing, MSE 408, MSE 412, or instructor's permission. An introduction to computer-aided manufacturing. Topics include basic issues in CAM and CIM; fixed, programmable, and flexible automation; introduction to numerical control technology; manual NC programming; the APT language; post-processing; graphical NC programming; NC programming with CAD/CAM systems; multiaxis part programming; and application areas. Two hours lecture.

409L. Fundamentals of Computer-Aided Manufacturing Laboratory (1)
Corequisite: COMP 409. One three hour laboratory per week.

410. Logic Programming (3)
Prerequisites: COMP 232, 282, 310. Programming techniques in the logic programming language PROLOG. Prenex conjunctive normal form and grammatical algorithms. Tableous, sequenzen, resolution, and other semi-decision procedures. Closures of relations, fixed point theory, control mechanisms, relationship to functional programming.

420. Advanced Operating System Concepts (3)
Prerequisites: COMP 322, 322L. An in-depth discussion of selected issues related to the study of operating systems. Areas

of coverage may include concurrency issues, resource allocation, storage management, and multiprocessor environments. Underlining theory and algorithms related to the issues will be discussed.

421. The UNIX Environment for Programmers (3)
Prerequisites: COMP 322, 322L. Usage of UNIX in a software development environment; rapid prototyping of large projects. Study available utilities, programming styles, efficiency issues, and operating system interfaces. Gain an advanced understanding of UNIX and its use in improving programmer productivity.

422. Advanced Computer and Operating System Architectures (3)
Prerequisites: COMP 322, 322L and EE 320. Study of interactions of computer and operating system architectures and the constraints they place on the operational capabilities and performance. Examination of evolutionary trends in computational architectures. This course is not available for graduate credit.

424. Computer System Security (3)
Prerequisite: COMP 322, 322L and 380. An analysis of the need for computer system security, and the security techniques in operating systems, data bases, and computer networks. Supporting techniques such as auditing, risk analysis, and cost-benefit tradeoffs are discussed.

426. Fault-Tolerant Software and Computing (3)
Prerequisites: COMP 322, 322L and 380. An examination of dependability requirements in computing, and the basic principles of system-level reliability and fault-tolerance. Software-based implementation of fault-tolerance in distributed systems. Fault-tolerant software and data bases: reliability modeling, fault-tolerance techniques (e.g., recovery blocks, N-version programming), and design approaches.

429. Computer Network Software (3)
Prerequisites: COMP 322, 322L, and MATH 340 or MATH 441. Basic software design and analysis considerations in networking computers into coherent, cooperating systems capable of processing computational tasks in a distributed manner. Network topology, routing procedures, message multiplexing and process scheduling techniques.

430. Language Design and Compilers (3)
Prerequisites: COMP 310, 380. An examination of the issues involved in the design and subsequent implementation of programming languages. Considerations of the implementation difficulties of including various features in a programming language. Tools and techniques to facilitate both the processing of programming languages and the building of programming processors.

431. Computer Modeling and Simulation (3)
Prerequisites: COMP 380 and MATH 340 or 441. A conceptual foundation for discrete event and continuous time simulation on computers is presented. Statistical considerations such as random number generation, design of experiments, output analysis and model correctness are considered. Programming in discrete event simulation languages such as GPSS, Simscript or SIMULA. Implementation issues for simulation languages.

432. Object-Oriented Programming (3)
Prerequisite: COMP 322, 322L. Principles of object-oriented design and programming; object-oriented languages such as Simula, C++ and Smalltalk will be compared to provide an understanding of the role of objects, methods, message passing, encapsulation, classes, inheritance and instance variables in a productive programming environment. Language design and programming issues will be discussed.

440. Database Design (3)
Prerequisite: COMP 380. Database structure including: structure definition, data models, semantics of relations, operation on data models. Database schemas: element definition, use and manipulation of the schema. Elements of implementation. Algebra of relations on a database. Hierarchical databases. Discussion of information retrieval, reliability, protection and integrity of databases.

450. Societal Issues in Computing (3)
Prerequisites: COMP 380 and senior standing. A survey course on the role of the digital computer in modern society. The dangers of the misuse of computers (as in the invasion of privacy), as well as the proper and intelligent use of the machines, are discussed. This course is not available for graduate credit.

461. Electronic Document Processing (3)
Prerequisites: COMP 380. Survey of current document processing tools. Document input with OCR. Design of editing, compression and encryption algorithms. Text and hypertext markup. Computer-assisted abstraction and translation. Design and implementation of document processing software.

465. Computer Graphic Systems and Design (2)
Prerequisite: COMP 380, MATH 262. Fundamental concepts of computer graphics. Graphics devices; graphics languages; interactive systems. Applications to art, science, engineering and business. Trade-offs between hardware devices and software support.

465L. Computer Graphic Systems and Design Laboratory (1)
Corequisite: COMP 465. One three hour laboratory per week.

467. Multimedia Systems Design (3)
Prerequisite: COMP 380 and Senior standing in Computer Science, or consent of instructor. Study of the fundamentals of multimedia storage, processing, communication, presentation, and display by digital means with emphasis on audio, still images and video media. It includes sampling theory, compression techniques and synchronization. Discussion of hypermedia and methodology issues. Multimedia

programming; software tools for authoring multimedia applications and interfaces.

467L. Multimedia Systems Design Laboratory (1)
Corequisite: COMP 467. One three hour laboratory per week.

469. Introduction to Artificial Intelligence (3)
Prerequisites: COMP 310, 380, 410. An exploration of the use of computers to perform computations normally associated with intelligence. These include game playing, theorem proving, problem solving, question answering and visual perception. Topics covered will include languages, system architectures and heuristic strategies for advanced, high level computations. Also covered will be computational models for knowledge representation, natural language and vision.

480. Software System Development (2)
Prerequisite: COMP 380. A project-oriented course to allow the students to apply their knowledge to the design of a large system. The student will identify a suitable computer problem, examine various methods of attacking it, choose a suitable one, and realize a solution in an appropriate computer language.

480L. Software System Development Laboratory (1)
Corequisite: COMP 480. One three-hour laboratory per week.

485. Human-Computer Interaction (3)
Prerequisite: COMP 380. The information exchange between humans and computer systems will be examined. Aspects of input/output devices, software engineering, and human factors will be discussed with respect to human-computer interactions. Topics include: text and graphic display; user modeling; program design, debugging, complexity and comprehension; and current research studies and methodologies.

487. Software Engineering with Ada (3)
Prerequisites: COMP 322, 322L, 380. Study of Ada's syntax and semantics as a programming language. The elements of object oriented programming and its application in the context of Ada will be discussed. Design of real-time software with Ada will be covered with examples and assignments. Emphasis will be placed on development of quality software with high reliability, portability and maintainability.

494. HCP. Honors Co-op (3-3)
Prerequisite: Second semester junior standing in major, upper division writing requirement, minimum 3.0 GPA. Coop Training Program. Supervised off-campus professional computing experience for selected Honors students. Honors Co-op units do not count toward General Education units or major requirements.

496. Experimental Courses in Computer Science (1-4)

499. Independent Study (1-3)

GRADUATE
(300-level courses in Computer Science do not carry credit for a masters' degree in Computer Science. 400-level courses required for the B.S. in Computer Science do not carry credit for the masters' degree.)

511. Robotics: Fundamentals and Applications (2)
(Same as MSE 511) Prerequisite: Consent of instructor. Introduction to the fundamentals of robotics, including motion and power, control, sensors, vision systems, and programming languages. Mathematical models, robot training, and the relationship to artificial intelligence will be considered. The application of robots and their integration into existing or proposed systems will be discussed. Students are expected to complete two projects as part of the course requirements.

511L. Robotics: Fundamentals and Applications Laboratory (1)
Corequisites: COMP 511. One three hour laboratory per week.

518. Algorithms and Data Structures (3)
This one course on programming and data structures covers all of the topics of COMP 110, COMP 110L and COMP 182. The laboratory involves programming design of significant projects in a high level programming language. This intensive course is open only to graduate students and cannot be used to satisfy the requirement at 30 units of approved graduate work.

518L. Algorithms and Data Structures Laboratory (1)
Corequisite: COMP 518. One three hour laboratory per week.

529. Advanced Network Topics (3)
Prerequisite: COMP 429. Advanced course on design and analysis of high-speed networks (Broadband ISDN and Asynchronous Transfer Mode [ATM] networks) and their protocols. Topics include: multimedia services integrating techniques including synchronous and asynchronous transfer modes. ATM standards. ATM switch architecture, ATM network traffic control, ATM experimental networks, high-speed LAN/MANs, internetworking with high-speed networks, and simulation techniques.

535. Parallel and Distributed Computing
Prerequisites: COMP 322, 322L. An examination of the issues necessary to program a set of computational resources to solve large, complex problems. Coverage of the issues inherent to parallel processing via a distributed programming model. Exposure to tradeoffs between communication and computation: scalability, concurrency, software engineering costs, etc.

560. Expert Systems (3)
Prerequisite: COMP 469 (COMP 485 desirable). Extensive introduction to the concepts and techniques of expert systems. Rationale for such systems including evaluation of prospective domains. Existing systems, those under development and likely future areas will be explored. Basic architecture is demonstrated using both example and rulebased systems. Commercial tools for building expert systems are surveyed and evaluated. Of

special concern will be knowledge acquisition methods. Guidelines for planning and managing development projects will be given.

565. Advanced Computer Graphics (2)
Prerequisites: COMP 322, 322L, or consent of instructor; COMP 465, 465L or equivalent is recommended. This course will cover the theory, design, implementation, and application of advanced computer graphics environments. Virtual Reality (VRML and immersive VR), 3D computer games, scientific visualization, and accelerated graphics APIs will be studied. The class requires a substantial software design and development effort by students.

565L. Advanced Computer Graphics Laboratory (1)
Corequisite: COMP 565. One three hour laboratory per week

585. Graphical User Interfaces (2)
Prerequisites: COMP 322, 322L, 380. The design and development of applications requiring graphical, direct manipulation, user interfaces will be examined using X Windows. The development of X Window applications will be compared with other graphical user interfaces (GUIs). Future GUIs, such as virtual realities, will be reviewed from the perspective of application design, development and development tools. Projects will require the use of advanced graphics workstations and provide GUI software design and development experiences.

585L. Graphical User Interfaces Laboratory (1)
Corequisite: COMP 585. One three hour laboratory per week

586. Object-Oriented Software Development (3)
Prerequisites: COMP 322, 322L, 380. Review of object oriented concepts. Comparison with functional methods. Benefits and pitfalls of object orientation. Fundamentals of object-oriented modeling -associations, links, states. Survey of object-oriented development methods. In-depth study of a current object-oriented method. Object-oriented software requirements analysis and modeling. Object-oriented preliminary design. Designing concurrent and multiprocessor systems. Object-oriented detailed design. Object-oriented and object-based implementations. Object-oriented testing.

588. Software Engineering Economics (3)
Prerequisite: COMP 380. An examination of the economic aspects of software development projects. Software development project variables, controls, analytic tools and the contexts of their use. Models of software development as educational as well as estimation tools. Software complexity compared to project accounting and managerial complexities. Current state-of-the art, research trusts, open questions in software economics.

595. Selected Topics (1-4)

609. Advanced Topics in CAD/CAM (3)
(Same as MSE 609) Prerequisites: MSE 408 and either MSE 603 or COMP 667, or equivalent with instructor's permission. Areas of current interest in Computer-Aided Design and Manufacturing. Topics include computer graphics software and hardware, mathematical basis of geometric modeling, data base management in manufacturing environments, user interface design considerations, and the CAD/CAM/CIM marketplace.

610. Data Structures and Algorithms (3)
Prerequisites: A grade of "B" or better in COMP 310, 380, and MATH 482, or demonstrate required level of proficiency. Topics include: design strategies for data structures and algorithms; theoretical limits to space and time requirements time/space trade offs; open problems in the field.

615. Advanced Topics in Computation Theory (3)
Prerequisites: A grade of "B" or better in COMP 310 and MATH 482, or demonstrate required level of proficiency. Languages and the theory of computation are studied in depth. Advanced material concerning regular and context free languages is covered. Deterministic context free languages, context sensitive languages, recursive and recursively enumerable sets are studied. Current areas of interest are investigated.

620. Computer System Architecture (3)
Prerequisites: A grade of "B" or better in COMP 322/322L or COMP 420, and EE 320N, or demonstrate required level of proficiency. Analysis and evaluation of individual computers, networks of computers and the programs which support their operation and use. Emphasis will be on comparison of architectures and the risks and benefits associated with various approaches and configurations.

630. Formal Semantics of Programming Languages (3)
Prerequisites: A grade of "B" or better in COMP 380 and COMP 310, or demonstrate required level of proficiency. Rigorous verification and formal proofs of correctness. Denotational semantics, models of axiomatic systems, fixpoint theory of computation. Soundness and completeness of programming logics. Abstract data types and other issues in the formal definition of programming languages.

667. CAD/CAM Systems Design (3)
Prerequisites: MSE 408 and COMP 465, 465L or equivalents with instructor's permission. Data structures and algorithms necessary to design and implement computer systems in manufacturing environments. Existing and anticipated technology will be discussed and evaluated. Students will design, implement, test, and evaluate CAD/CAM systems by building upon standard computer graphics packages.

680. Software Engineering (3)
Prerequisite: A grade of "B" or better in COMP 380, or demonstrate required level of proficiency. An examination of the critical theoretical problems underlying the specification, design, development and evaluation of large software systems,

and the extent to which existing techniques and methodologies cope with these problems.

695. Selected Topics (1-4)

696. Directed Graduate Research (2-3)
For Credit/No Credit only.

698. Thesis or Graduate Project (2-6)

699. Independent Study (1-3)

EXPERIMENTAL AND SPECIAL TOPICS COURSES

296JAV. Java Programming (2)
Prerequisite: Comp 101/105 or 106/L or 110/L equivalent. Corequisite: Comp 296JAL. Intended for students with prior experience in high level computer programming. Instruction and practice in the Java programming language with coverage of Object-Oriented Programming concepts, the Java Foundation Classes (SWING/AWT), event handling, applets, multithreading and network applications. Two hours lecture and one three-hour lab per week. Lab projects involve both micro computers and UNIX-based main frame access. (Not recommended for Computer Science majors who complete Comp 110/L at CSUN.).

296JAL. Java Programming Laboratory (1)
Corequisite: COMP 296JAV. One three hour laboratory per week.

396SE. Introduction to Software Engineering Projects (2)
Prerequisites: COMP 222, 232, and either COMP 242 or 282. Concepts and techniques for systems engineering, requirements analysis, design implementation and testing of large-scale computer systems. Principles of software engineering for production of reliable, maintainable and portable software products. Emphasis on functional analysis and structured design techniques. Topics include unit integration and systems testing, configuration management, and software quality assurance practices. Participation in group activities involving analysis, design and implementation of a software intensive system. Introduction to Computer Aided Software Engineering (CASE). This is the lecture portion of the first semester of a one year course in software engineering involving the design and development of a complete software system as a group project. The project begun in this course will be completed in COMP 496SE/SEL in the following semester. Satisfactory completion of COMP 396SE/SEL will satisfy the COMP 380 requirement for Computer Science majors.

396SEL. Intro to Software Engineering Projects Lab (1)
Corequisite: COMP 396SE. One three hour laboratory per week.

496ECT. E-Commerce Technologies (3)
Pre-requisite: COMP 322, 322L, 380. Overview of developments in Internet based e-commerce. Study of hardware, software, and network technologies for implementing e-commerce systems. Software and systems architectures for different types of e-commerce systems: business to consumer, business to business, online auctions, and exchanges. Design and implementation of virtual storefronts. Connectivity to backend databases, legacy systems and third party solution providers for electronic payments, authentication and fulfillment services. Security including use of firewalls and encryption technologies. Survey of emerging technologies.

496MC. Multimedia Computing (3)
Prerequisite: Comp 380 and Senior standing in Computer Science, or consent of instructor. Introduction to digital media elements and processing . Study of sampling theory, compression techniques, digital representation and storage. Application of multimedia programming and authoring tools. Exploring multimedia computing systems including hypermedia and methodology issues.

496SE. Advanced Software Engineering Projects (2)
Prerequisites: COMP 396SE/SEL or 380 and the consent of the instructor. A project-oriented course to allow students to apply their knowledge of software engineering to the design of a large software system. This is the lecture portion of the second semester of a one year course in software engineering involving the design and development of a complete software system as a group project. The project begun during the previous semester in COMP 396SE/SEL will be completed in this course. Satisfactory completion of COMP 496SE/SEL is equivalent to taking COMP 480/L.

496SEL. Advanced Software Engineering Projects Lab (1)
Corequisite: COMP 496SE. One three hour laboratory per week.

595ADB. Advanced Database Systems (3)
Prerequisite: COMP 440 or demonstrate required level of proficiency in database systems. Topics include: on-going trends in database systems, emphasizing database fundamentals and new research directions; study a collection of papers that have influenced the field and represented the state of the art in database systems such as Bench-marking, Query Optimization, Data Warehousing, Data Mining, Transaction Processing, Concurrency Control and Recovery, Object-Oriented (OO) and Object-Relational DB, Distributed and Client-Server, and Parallel DB.

595SEC. Advanced Computer System Security (3)
Prerequisite: COMP 424. An advanced seminar style course covering computer system security technology, protocols, and practices. It includes in-depth study and discussion of the following topics: applied cryptography; common attack methods such as covert channels, Trojan horses, and viruses; protection on operating systems including security kernels and trusted computing bases; data base security; multilevel security in networks and distributed systems; the administration of security in computer systems; and legal and ethical issues.