INTRODUCTION

The undergraduate computer science program at Cal State Fullerton offers students a comprehensive foundation that will permit them to adapt to new technologies and new ideas. The program spans a wide range, from its theoretical and algorithmic foundations to cutting-edge developments in bioinformatics, communications systems, computer security, databases, digital game design, intelligent systems, software engineering, and other exciting areas.

The program provides students with a comprehensive background to take on varied categories of work. They are offered the necessary theories, principles and practices to design and implement software that permits them to take on challenging programming jobs. They have the opportunity to become well-equipped to devise new ways to use computers. The theoretical foundations available in the program provide the background to help develop effective ways to solve computing problems. This background allows students to determine the best possible ways to store information in databases, send data over networks, and display complex images.

The Bachelor of Science degree in Computer Science is accredited by the Computing Accreditation Commission of ABET, www.abet.org.

LEARNING GOALS AND STUDENT LEARNING OUTCOMES

The following learning goals and student learning outcomes have been established for students pursuing a degree in Computer Science:

Program Educational Objectives
A. Technical Growth – Graduates will be successful in modern computing practice, integrate into the local and global workforce, and contribute to the economy of California and the nation
B. Professional Skills – Graduates will continue to demonstrate the professional skills necessary to be competent employees, assume leadership roles, and have career success and satisfaction
C. Professional Attitude and Citizenship – Graduates will become productive citizens with high ethical and professional standards, who make sound technical or managerial decisions, and have enthusiasm for the profession and professional growth

Student Outcomes
• Able to apply knowledge of computing and mathematics appropriate to the discipline
• Able to analyze a problem, and identify and define the computing requirements appropriate to its solution
• Able to apply mathematical foundations, algorithmic principles, and computer science theories in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices
• Able to design, implement and evaluate a computer-based system, process, component, or program to meet desired needs
• Able to apply design and development principles in the construction of software systems of varying complexity
• Able to function effectively on teams to accomplish a common objective
• Able to communicate effectively with a range of audiences
• Recognize the need for and able to engage in continuing professional development
• Able to use current techniques, skills, and tools necessary for computing practice
• Able to analyze the local and global impact of computing on individuals, organizations, and society
• Demonstrate an understanding of professional, ethical, legal, security and social issues and responsibilities

BACHELOR OF SCIENCE IN COMPUTER SCIENCE (124 UNITS)

The degree program for the Bachelor of Science in Computer Science assumes that students have already obtained a working knowledge of personal computing fundamentals and applications, including word processing, spreadsheets, database systems, e-mail systems and presentation graphics.

Computer Science Placement Examination

The curriculum begins with a three-course sequence covering concepts of programming and data structures. Students may have knowledge of these topics, but do not have the courses to transfer, nor AP scores to submit; thus, may take the Computer Science Placement Examination to waive one or more of these courses. The test may be taken only once, and scores are valid for two consecutive semesters.

Computer Science Core (46 units)

Lower-Division Core (18 units)

CPSC 120  Introduction to Programming (3)
CPSC 121  Programming Concepts (3)
CPSC 131  Data Structures Concepts (3)
CPSC 223H  Visual Basic Programming (3)
   OR CPSC 223J  Java Programming (3)
   OR CPSC 223N  C# Programming (3)
CPSC 240  Computer Organization and Assembly Language (3)
CPSC 254  UNIX and Open Source Systems (3)

Upper-Division Core (28 units)

Students must take and pass the Examination in Programming Proficiency (EPP) before taking most upper-division Computer Science courses. Students who do not pass the EPP will be required to take CPSC 301.

CPSC 311  Technical Writing for Computer Science (3)
CPSC 315  Social and Ethical Issues in Computing (1)
CPSC 323  Programming Languages and Translation (3)
CPSC 332  File Structures and Database Systems (3)
CPSC 335  Problem Solving Strategies (3)
CPSC 351  Operating Systems Concepts (3)
CPSC 362  Foundations of Software Engineering (3)
CPSC 440  Computer System Architecture (3)
CPSC 471  Computer Communications (3)
CPSC 481  Artificial Intelligence (3)

Elective Track Requirements (15 units)

Select an Elective Track to support specific career goals.

Multimedia and Digital Game Technologies

CPSC 386, 484, 486, 489

Plus any three units of adviser-approved, upper-division CPSC courses

Internet and Enterprise Computing Technologies

CPSC 431, 473, 476

Plus any six units of adviser-approved, upper-division CPSC courses

Software Engineering

CPSC 462, 464; 463 OR 466

Plus any six units of adviser-approved, upper-division CPSC courses

Scientific Computing

Completing the Mathematics courses listed below also meets the requirements for a minor in Mathematics.

MATH 250A, 250B, 340, 370

Plus any three units of adviser-approved 400-level CPSC courses

Custom

With the approval of an academic adviser, students may develop a track based on their career goals or specific academic interests or specific themes. A custom track consists of upper-division Computer Science or related courses. At least nine units must be 400-level Computer Science courses with no more than three units selected from courses numbered 490–499. In addition to courses already listed in the other tracks, students may also include the following courses:

CPSC 303, 322L, 376, 433, 459, 477, 483, 485, 491T, 495, 499

Mathematics Requirements (18 units)

MATH 150A,B  Calculus (4,4)
MATH 270A,B  Mathematical Structures (3,3)
MATH 338  Statistics Applied to Natural Sciences (4)
Science Requirements (12 units)

**Physical Science (8 units)**

Choose one of the following:

**Physics**

PHYS 225/225L Fundamental Physics: Mechanics/Laboratory (4)
PHYS 226/226L Fundamental Physics: Electricity and Magnetism/Laboratory (4)

**Chemistry**

CHEM 120A General Chemistry (5)
CHEM 125 General Chemistry for Engineers (3)

**Geology**

GEOL 101/101L Physical Geology/Laboratory (4)
GEOL 201/201L Earth History/Laboratory (4)

**Biological Science (4 units)**

BIOL 101 Elements of Biology (3)
BIOL 101L Elements of Biology Laboratory (1)

**General Education**

Because of high unit requirements for a major in Computer Science, there is a six-unit exemption in General Education (G.E.). Students are strongly urged to consult with an adviser in the Academic Advising Center, UH-123 to help develop their G.E. program.

**Minimum Academic Requirements**

A "C-" (1.7) or higher is required in all courses applied to the core. Exception: up to six units of credit with grades in the range of "D-" (0.7) through "D+" (1.3) may be earned in elective tracks, mathematics and science courses only. A mathematics course taken to satisfy GE category B.4 and the upper-division writing course (CPSC 311) must be passed with a minimum grade of "C".

**MINOR IN COMPUTER SCIENCE (15 UNITS)**

CPSC 120 Introduction to Programming (3)
CPSC 121 Programming Concepts (3)
CPSC 131 Data Structures Concepts (3)
CPSC 313 The Computer Impact (3)

Three units of adviser-approved, upper-division courses

**MASTER OF SCIENCE IN COMPUTER SCIENCE (30 UNITS)**

**Admission to Graduate Standing – Conditionally Classified**

Students must meet the CSU requirements for admission to a master’s degree program. Please consult the Graduate Admissions section in this catalog for complete information. Any deficiencies must be made up.

**Classified Graduate Standing**

Achievement of this status requires the following:

1. approval of a formal study plan (see description below) by the Computer Science Graduate Committee and the Associate Vice President for Graduate Studies and Research (or designee);
2. satisfactory completion of no more than nine units on the study plan;
3. satisfactory completion of the following courses or equivalents, including prerequisites – CPSC 121, 131, 240, 323, 335, 351, 362 and MATH 270A,B, 338; and
4. competency in written communication in English must be demonstrated by a passing score on the California State University Examination in Writing Proficiency. The requirement must be satisfied before the student can be classified and before 500-level courses may be attempted. The student who fails to pass the EWP test may complete CPSC 311 with a “B” (3.0) or better as an alternative to the EWP requirement.

The courses provide an adequate basis for graduate work, while not unfairly precluding admission of persons without a bachelor’s degree in computer science. These courses have prerequisites, and students without preparation in a closely related degree may have additional work to complete. Refer to the catalog descriptions for prerequisites of each course deficiency.

Students with knowledge equivalent to any or all of these prerequisite courses are encouraged to satisfy such prerequisites by advanced placement examinations. Consult a Computer Science graduate adviser for further information.

**Study Plan**

Prior to admission to classified graduate standing in Computer Science, each student, with the aid of a Computer Science graduate adviser, shall prepare and submit for approval by the Computer Science Department graduate committee a formal study plan consisting of 400-level and graduate coursework.

This shall include CPSC 440, 462, 589, 597 or 598; one of 541, 542, 543, 544, 545, 546, 547 or 548; and 15 units of electives (nine units must be at the 500 level). At least 15 units shall represent courses offered by the Department of Computer Science. Courses offered by other disciplines, not listed here, and related to the student’s objectives in Computer Science may be approved by petition to the Department of Computer Science.

All coursework in the study plan must be completed with a GPA of at least 3.0.

**Graduate Student Advisement**

The graduate program adviser provides overall supervision of the graduate program. The individual student chooses an adviser for the thesis or project from the Computer Science Department’s full-time faculty on the basis of the student’s particular interests and objectives.
COMPUTER SCIENCE COURSES

Courses are designated as CPSC in the class schedule. Prerequisites for Computer Science courses may be waived only by department petition.

103 Introduction to Personal Computer Applications (3)
Introduction to use and application of personal computers: word processing, spreadsheets, database systems, e-mail systems and World Wide Web. Evaluation of personal computers and software. (2 hours lecture, 2 hours laboratory)

120 Introduction to Programming (3)
Corequisite: MATH 125. Introduction to the concepts underlying all computer programming: design and execution of programs; sequential nature of programs; use of assignment, control and input/output statements to accomplish desired tasks; use of functions and arrays. Structured programming methodologies. (1.5 hours lecture, 3 hours laboratory)

121 Programming Concepts (3)
Prerequisite: CPSC 120 or passing score on Computer Science Placement Exam. Structure of algorithms; functions; strings and data types; pointers and linked structures; classes and objects; recursion; inheritance; polymorphism; exception handling; documentation. Object-oriented programming methodology. (2 hours lecture, 2 hours laboratory)

131 Data Structures Concepts (3)

223H Visual BASIC Programming (3)
Prerequisite: CPSC 131. Elements of Visual BASIC, forms and controls, properties, mouse events, multiple-document interface, processing files, accessing databases, dynamic data exchange, object linking and embedding. (2 hours lecture, 2 hours laboratory)

223J Java Programming (3)
Prerequisite: CPSC 131. Characteristics of Java: portable, robust, secure, object-oriented, high performance; using the Java environment; server administration; types, expressions, and control flow; classes, interfaces, and packages; threads; exceptions; class libraries; Java for the Internet; tools, the Java Virtual machine. (2 hours lecture, 2 hours lab per week)

223N C# Programming (3)
Prerequisite: CPSC 131. Characteristics of C#, object-oriented design concepts, control structures, methods, arrays, classes, objects, inheritance, polymorphism, exception handling, graphical user interfaces, multithreading, characters, strings, files, streams. Rudiments of the Unified Modeling Language. Software development assignments. (2 hours lecture, 2 hours laboratory)

240 Computer Organization and Assembly Language (3)
Prerequisites: CPSC 131 and MATH 270A or 280. Digital logic and architecture of a computer system, machine level representation of data, memory system organization, structure of low-level computer languages; machine, assembly and macro language programming; principles of assembler operation, input-output programming, interrupt-exception handling. Laboratory programming assignments. (2 hours lecture, 2 hours laboratory)

253U Workshop in UNIX (1)
Prerequisite: CPSC 121 or EGME 205. Workshop in the use of the UNIX operating system. (2 hours activity)

254 Software Development with Open Source Systems (3)
Prerequisite: CPSC 131. Philosophy of open source software development. Open source operating systems, such as Linux; open source development tools; open source programming languages, such as Python; open source software development processes; open source software licensing issues. (2 hours lecture, 2 hours laboratory)

301 Programming Lab Practicum (2)
Prerequisites: CPSC 131 and 253U or 254. Intensive programming covering concepts learned in lower-division courses. Procedural and object oriented design, documentation, arrays, classes, file input/output, recursion, pointers, dynamic variables, data and file structures. Credit/no credit only.

303 Multimedia Concepts (3)
Prerequisites: CPSC 121 and completion of the General Education (G.E.) critical thinking requirement. Components and issues associated with multimedia technology, applications of multimedia and its evolution. Laboratory activities include developing a multimedia application using a PC-based authoring tool. (2 hours lecture, 2 hours laboratory)

311 Technical Writing for Computer Science (3)
Prerequisite: ENGL 101, CPSC 131. Practice in developing documentation skills as used in the computer field. Topics include proposals, feasibility studies, user guides and manuals, business communication and technical presentation. Case studies in professional ethics. Written and oral reports required.

313 The Computer Impact (3)
Prerequisites: junior or senior standing and one course from G.E. Category B.4. Effect of computer use on individuals and organizations. Side effects of innovative technology and the resulting changes to organizations, social institutions, and human perceptions of events. Personal responsibility, legal ramifications and educational implications. Hands-on use of e-mail and the World Wide Web.
315 Social and Ethical Issues in Computing (1)
Prerequisite: CPSC 311. Relevant issues that responsible professionals will face in a complex technological society. Professional ethics, computer control, piracy, encryption, benefits and downside of computers, privacy and computer crimes. Written and oral reports required.

322L Introduction to Computer Aided Design (3)
(Same as EGM 322L)

323 Programming Languages and Translation (3)
Prerequisites: Examination in Programming Proficiency. Basic concepts of programming languages and principles of translation. Topics include history of programming languages, various programming paradigms, language design issues and criteria, development of practical translators for modern programming languages.

332 File Structures and Database Systems (3)
Prerequisite: CPSC 131. Fundamental theories and design of database systems, the Structured Query Language (SQL), basic concepts and techniques of data organization in secondary storage. Topics include introduction to database systems, ER model, relational model, index structures, and hashing techniques.

335 Algorithm Engineering (3)

351 Operating Systems Concepts (3)
Prerequisite: CPSC 253U or 254. Corequisite: Examination in Programming Proficiency or CPSC 301. Resource management, memory organization, input/output, control process synchronization and other concepts as related to the objectives of multi-user operating systems.

362 Foundations of Software Engineering (3)
Prerequisites: CPSC 311 and Examination in Programming Proficiency. Basic concepts, principles, methods, techniques and practices of software engineering. All aspects of the software engineering fields. Computer-Aided Software Engineering (CASE) tools are used.

376 Client/Server Systems with Java (3)

386 Introduction to Game Design and Production (3)
Prerequisite: CPSC 131. Current and future technologies and market trends in game design and production. Game technologies, basic building tools for games and the process of game design, development and production.

431 Database and Applications (3)
Prerequisite: CPSC 332. Database design and application development techniques for a real-world system. Topics include system analysis, requirement specifications, conceptual modeling, logic design, physical design and web interface development. Develop projects using contemporary database management system and web-based application development platform.

433 Data Security and Encryption Techniques (3)
Prerequisites: CPSC 311, 351, MATH 270B. System security and encryption. Current issues in security, encryption and privacy of computer based systems.

440 Computer System Architecture (3)
Prerequisite: CPSC 240. Computer performance, price/performance, instruction set design and examples. Processor design, pipelining, memory hierarchy design and input/output subsystems.

451 Advanced Operating Systems (3)
Prerequisite: CPSC 351. Internal structures of a modern operating system. Specific topics include processes, process communication, file systems, networking, and the I/O system. There will be several programming assignments which will utilize calls and other low-level interfaces.

459 Micro-Computer Software Systems (3)
Prerequisite: CPSC 351. Design and implementation of software. Analysis of a micro-computer operating system and working on a team to implement a significant programming assignment.

462 Software Design (3)
Prerequisite: CPSC 362. Concepts of software modeling, software process and some tools. Object-oriented analysis and design and Unified process. Some computer-aided software engineering (CASE) tools will be recommended to use for doing homework assignments.

463 Software Testing (3)
Prerequisite: CPSC 362. Software testing techniques, reporting problems effectively and planning testing projects. Students apply what they learned throughout the course to a sample application that is either commercially available or under development.

464 Software Architecture (3)
Prerequisite: CPSC 362. Basic principles and practices of software design and architecture. High-level design, software architecture, documenting software architecture, software and architecture evaluation, software product lines, and some considerations beyond software architecture.
466 Software Process (3)
Prerequisite: CPSC 362. Practical guidance for improving the software development and maintenance process. How to establish, maintain and improve software processes. Exposure to some common process models, such as CMM, CMMI, PSP and TSP.

471 Computer Communications (3)
Prerequisite: CPSC 351. Introduction to digital data communications. Terminology, networks and their components, common-carrier services, telecommunication facilities, terminals, error control, multiplexing and concentration techniques.

473 Web Programming and Data Management (3)
Prerequisite: CPSC 332. Various techniques for developing Web-based database applications using software engineering methodology. Introduce concept and architecture of Web servers, Web database design techniques, client/server side programming, and Web applications tools and techniques.

474 Distributed Computing Using Web Service and .NET Remoting (3)
Prerequisite: CPSC 473. Concepts of distributed computing and Web services, the applications of XML and Web services, distributed applications development techniques with Web services and .NET Remoting.

476 Java Enterprise Application Development (3)
Prerequisites: CPSC 223J, 351. Concepts and architecture of enterprise applications, components, services and communication technologies, Dependency injection, application tiers, remote objects, distributed transactions, message queues, web services and object-relational mapping. Enterprise application development in Java with build tools, containers and application servers.

477 Introduction to Grid Computing (3)
Prerequisite: CPSC 351. Introduction to various aspects of grid computing emphasizing integration of homogeneous and inhomogeneous computational resources to provide high-performance computing seamlessly, efficiently and securely; using Globus toolkit as the integration framework for demonstrating and implementing various aspects of grid computing.

481 Artificial Intelligence (3)
Prerequisite: CPSC 335. Use of computers to simulate human intelligence. Topics include production systems, pattern recognition, problem solving, searching game trees, knowledge representation and logical reasoning. Programming in AI environments.

483 Data Mining and Pattern Recognition (3)
Prerequisite: CPSC 335. Classification techniques, discriminant functions, training algorithms, potential function theory, supervised and unsupervised learning, feature selection, clustering techniques, multidimensional rotations and rank ordering relations.

484 Principles of Computer Graphics (3)
Prerequisites: Examination in Programming Proficiency and MATH 150B, 270B. Examination and analysis of computer graphics, software structures, display processor organization, graphical input/output devices, display files. Algorithmic techniques for clipping, windowing, character generation and viewpoint transformation.

485 Computational Bioinformatics (3)
Prerequisites: junior or senior standing, BIOL 101, CPSC 131. Algorithmic approaches to biological problems. Specific topics include motif finding, genome rearrangement, DNA sequence comparison, sequence alignment, DNA sequencing, repeat finding and gene expression analysis.

486 Game Programming (3)
Prerequisites: CPSC 386, 484. Data structures and algorithms used for real-time rendering and computer game programming. Build upon existing mathematics and programming knowledge to create interactive graphics programs.

489 Game Development Project (3)
Prerequisite: CPSC 486. Individually or in teams, student design, plan and build a computer game.

491T Variable Topics in Computer Science (1-3)
Prerequisites: junior or senior standing and consent of instructor. Lectures and/or workshop covering various current Computer Science topics. Course may be repeated for up to 3 units. Course topics may be taken only once.

495 Internship in Computer Science (1-3)
Prerequisites: Computer Science or related major and consent of instructor. Practical experience relevant to computer science in government or private agencies. Written and oral reports required.

499 Independent Study (1-3)
Prerequisite: approval by the computer science chair. Special topic in Computer Science, selected in consultation with and completed under the supervision of instructor.

531 Advanced Database Management (3)
Prerequisite: CPSC 431. Implementation techniques for query analysis, data allocation, concurrency control, data structures, and distributed databases. New database models and recent developments in database technology. Student projects directed to specific design problems.

541 Systems and Software Standards and Requirements (3)
542 Software Verification and Validation (3)
Prerequisite: CPSC 362 or equivalent work experience. How to ensure that a high quality software product is developed. Theory and practice of software verification and validation (V&V), such as Software integrity levels, Minimum V&V tasks for each software integrity level, walkthroughs, inspections and Cleanroom. Software testing topics: white- and black-box testing, boundary value analysis, equivalence class partitioning, unit testing, functional testing and how to create test plans.

543 Software Maintenance (3)
Prerequisite: CPSC 362 or equivalent work experience. Theory and practice of maintaining large-scale software and how to construct maintainable software. Maintenance framework, along with maintenance process, process management and maintenance measures. Topics include fundamentals of software change, implications of software change, maintenance process models, program understanding, reusability for maintenance, reverse engineering, maintenance testing, software configuration management and tools in maintenance.

544 Advanced Software Process (3)
Prerequisite: CPSC 362 or equivalent work experience. Advanced guidance for defining and improving the software development process. Concepts of software maturity framework, principles of process improvement and software process assessment. Current topics such as CMMI and SCAMPI.

545 Software Design and Architecture (3)
Prerequisite: CPSC 362 or equivalent work experience. Advanced software design and architecture principles focusing a software engineering approach to the development process. Topics include architecture business cycle, quality attributes, attribute-driven design method, architectural styles, design patterns, software product lines and component-based design.

546 Modern Software Management (3)
Prerequisite: CPSC 362 or equivalent work experience. Modern project management methodologies and techniques. Software development process. Planning, estimating, organizing, directing, monitoring, controlling software projects and managing risks. Other related software management issues, such as infrastructure, quality software development, project and product metrics, and external factors.

547 Software Measurement (3)
Prerequisite: CPSC 362 or equivalent work experience. Current software measurement practices. Topics include: how to establish an effective software metrics program in a software organization; how to measure software product, project and process; how to apply Statistical Process Control and other statistical techniques in software development process. High maturity concepts defined in CMMI model will be discussed. Stresses a practitioner-based approach.

548 Professional, Ethical and Legal Issues for Software Engineers (3)
Prerequisite: CPSC 362 or equivalent work experience. Professional, legal and ethical issues pertaining to software engineering. Topics include professional codes of ethics, intellectual property laws, computer privacy and human-computer interaction. Relevant regulatory documents and their applications.

551 Operating Systems Design (3)
Prerequisite: CPSC 351. Design and evaluation techniques for controlling automatic resource allocation, providing efficient programming environments and appropriate user access to the system, and sharing the problem solving facilities.

558 Advanced Computer Networking (3)
Prerequisite: CPSC 471. System-oriented view of computer network design, protocol implementation, networking, high-speed networking, network management, computer network performance issues.

566 Advanced Computer Graphics (3)
Prerequisite: CPSC 484. Three-dimensional: reflection models, shading techniques, rendering process, parametric representation, ray tracing, radiosity, texture, anti-aliasing, animation, color science.

572 Survey of Pharmaceutical and Medical Devices Technology (3)
Prerequisites: enrollment in the Master of Biotechnology (MBt) degree program or consent of instructor, and MGMT 540. Corequisites: BIOL/CHEM 570, BIOL/MATH 571. Technologies involved in developing drug and medical devices, factors considered in designing medical devices, characteristics of good drug manufacturing practices and validation processes necessary to meet regulatory requirements. Students will work collaboratively to solve problems. (Same as BIOL 572)

583 Expert Systems Design Theory (3)
Prerequisite: CPSC 481. Knowledge representation and search strategies for expert systems; logic programming; expert system tools. Project.

585 Artificial Neural Networks (3)
Prerequisite: CPSC 481. Principles of neural networks; neural networks paradigms, software implementations, applications, comparison with statistical methods, use of fuzzy logic; project.

589 Seminar in Computer Science (3)
Prerequisites: one 400-level course in Computer Science and passing score on the Examination in Writing Proficiency. Research methods in computer science. Student presentations covering current topics, research advances, updating of concepts and verifications of principles of computer science. (Examples: large-scale parallelism, Internet security, design for user interfaces, computers in instruction).
597 Project (3)
Prerequisites: classified graduate standing, approval of the computer science graduate adviser and CPSC 589.

598 Thesis (3)
Prerequisites: classified graduate standing, approval of the computer science graduate adviser and CPSC 589.

599 Independent Graduate Research (1-3)
Prerequisites: classified graduate standing, approval of the computer science department chair and CPSC 589. Special topic in computer science, selected in consultation with and completed under supervision of a full-time faculty member.