

Computer Science

(See “Computer Information Systems” section for additional computer courses.)

Degrees and Certificates Awarded

Associate in Science Degree, Computer Science
 Certificate of Achievement, Computer Science
 Skills Competency Award, Web Programming
 Skills Competency Award, Mobile Application Developer

Program Description

We are in the Computer Age. Virtually every occupation in the world today has an interface with computers. From the microprocessor under the hood of your automobile to the larger scale systems used by Congress to formulate new laws, we all are affected in our daily lives by computers. Never before in history has any single endeavor grown so fast or become so universally accepted.

At Santa Barbara City College, we are helping students meet the challenge presented by this new technology. Classes, from introductory to advanced topics, are designed to provide General Education, transfer and occupational training. The AS Degree requirements to follow are designed to prepare students for employment or for transfer to both the CSU and UC systems.

Program Student Outcomes

1. Decompose problems into algorithms.
2. Create programs, using at least three languages.
3. Create programs that use flow control and looping constructs (e.g., for and while).
4. Create programs that utilize standard data structures (e.g., queues and lists).
5. Create programs that use object-oriented concepts.
6. Create programs, using current programming environments.

Department Offices

Computer Center (H-245, ext. 2401/2402)
 Stephen Strenn, *Chair* (H-226, ext. 2490)
 Soheyla Javanbakht, *Lab Teaching Assistant*
 James Howard, *Lab Teaching Assistant*
 TBD, *Dean*

Faculty and Offices

Stephen Strenn (H-226, ext. 2490)
 Salmaun (Sam) Massooman (H-226, ext. 2452)
 Jacqueline Kuehn (H-213, ext. 2693)
 Dean Nevins (H-214, ext. 2791)

AS Computer Science Degree Requirements

Department Requirements (33-37 units)

CS 105 — Theory and Practice I.....	3
CS 106 — Theory and Practice II.....	3
CS 107 — Computer Architecture and Organization.....	3
CS 108 — Discrete Structures	4
CS 140 — Object-Oriented Prog, Using C++	4
MATH 150 — Calculus with Analytic Geom. I.....	5
PHYS 121 — Mechanics of Solids and Fluids.....	5

Plus at least 6 units from the following:

CS 104 — Introduction to Programming.....	4
CS 111 — HTML and Webmastering.....	3
CS 115 — JavaScript and Dynamic HTML.....	3
CS 116 — Webserver Programming	3
CS 120 — Java Programming.....	3

Plus two courses from the following:

CS 123 — Android Programming	1.5
CS 130 — Intro. to the Linux Operating System	2.5
CS 132 — Digital Logic Design	3
CS 133 — Intro to Programming for Engineers.....	3
CS 137 — C Programming	3
CS 165 — Software Design Patterns	1.5
CS 180 — Software Engineering with UML.....	3
CS 187 — iOS Programming	3
CS 189 — Programming Practicum	1
MATH 160 — Calculus with Analytic Geom II.....	5
MATH 200 — Multivariable Calculus	4
MATH 210 — Linear Algebra.....	4
MATH 220 — Differential Equations	4
PHIL 205 — Introduction to Logic.....	3
PHYS 122 — Electricity and Magnetism	5

**NOTE: PHYS 102 does not count toward department requirement if either PHYS 121 or 122 has been taken. MATH 250/260 may also be count toward the elective requirement. A course used to satisfy one requirement may not be used to satisfy another requirement (double-counting is not allowed).*

College Requirements

For complete information, see “Graduation Requirements” in the *Catalog* Index.

Certificate of Achievement: Computer Science

Department Requirements (32.5-35.5 units)

CS 101 — Computer Concepts	4
CS 105 — Theory and Practice	3
CS 120 — Java Programming	3
CS 130 — Introduction to the Linux Operating System.....	2.5
CS 137 — C Programming or	3
CS 140 — Object-Oriented Programming Using C++	4
MATH 107 — Intermediate Algebra or	4
MATH 111 — Intermediate Algebra for Math, Science and Business Majors	5
PHIL 205 — Introduction to Logic	3
PHYS 102 — Introduction to Physics for Science Majors	4

Plus two courses from the following:

CS 111 — HTML and Webmastering.....	3
CS 106 — Theory and Practic II.....	3
CS 107 — Computer Architecture and Organization.....	3
CS 137 — C Programming.....	3
CS 140 — Object-Oriented Programming Using C++	4

Note: *A course used to satisfy one requirement may not be used to satisfy another requirement (double-counting is not allowed).*

Students must complete all department requirements for the certificate with a cumulative GPA of 2.0 or better. Candidates for a Certificate of Achievement are required to complete at least 20% of the department requirements through SBCC.

Skills Competency Award: Web Programming

Department Requirements (16-16.5)

CIS 230 — Active Server Pages Using Visual Basic or	4
CS 125 — C# Programming and	3
CS 127 — ASP.NET Using C#	1.5

CS 111 — HTML and Webmastering.....	3
CS 115 — JavaScript and Dynamic HTML.....	3
CS 116 — Web Server Programming.....	3
CS 120 — Java Programming.....	3

Students must complete the above courses with a grade of “C” or higher or credit in all courses.

**Skills Competency Award:
Mobile Application Developer**

Department Requirements (10.5-13.5)

CS 111 — HTML and Webmastering.....	3
CS 120 — Java Programming or	3
CS 125 — C# Programming.....	3
CS 126 — Microsoft Mobile Device Programming	1.5
CS 127 — ASP.net Using C# or	1.5
CS 116 — Web Server Programming or	3
CS 187 — iPhone and iPod Touch Programming	3

Students must complete the above courses with a grade of “C” or higher or credit in all courses.

Sample Program

To satisfy the course requirements of the Computer Science major, the student is encouraged to meet with a member of the Computer Science faculty for individualized guidance. A suggested course sequence for Computer Science courses follows:

First Year	Second Year
<i>First Semester</i>	<i>Third Semester</i>
CS 101	CS 120
	CS 131
 <i>Second Semester</i>	 <i>Fourth Semester</i>
CS 130	CS 140
CS 135	CS 145J or
	CS 145P

Preparation for Transfer

Course requirements for transfer vary depending upon the college or university a student wishes to attend. Because Computer Science is such a competitive major at many four-year schools, it is *most important* for a student to consult with his/her counselor and departmental adviser before planning an academic program for transfer. Information sheets for majors,

outlining transfer requirements, are available in the Counseling Center.

Program Cost and Outcome

For planning purposes, the webpage below provides information on the cost of attendance, program length (assuming a student attends full time), financing options and historical student completion rates: www.sbcc.edu/financialaid/gainfulemployment/Computer%20Science.htm

Computer Science Courses

CS 101 — Computer Concepts

(4) — CSU, UC*

Skills Advisories: MATH 100 and Eligibility for ENG 103

Hours: 72 lecture

Survey of the concepts of computer hardware and software, with emphasis on the latest technologies and programming. Topics include, but are not limited to, the Internet, productivity applications, databases, programming and numbering systems. Suitable for all majors and is a recommended first course for Computer Science majors. (*UC transfer limit: CS 101 combined with CIS 101: maximum credit, one course.)

CS 104 — Introduction to Programming

(3) — CSU, UC

Skills Advisories: Eligibility for ENG 100 and 103

Hours: 90 (36 lecture, 54 lab)

Exposes students to the fundamental concepts of programming using an object-oriented language and is intended as an introductory-level programming course. Ideal for beginners and also serves as the first course in a sequence that is compliant with the standards of the Association for Computing Machinery.

CS 106 — Theory and Practice II

(3) — CSU, UC

Prerequisites: CS 105 with a minimum grade of "C"

Skills Advisories: Eligibility for ENG 103

Hours: 90 (36 lecture, 54 lab)

Study of data structures and algorithms. Design, coding and testing of linked lists, trees, queues, stacks, hash tables, and other dynamic data structures, as well as searching and sorting algorithms. Time and space analysis of data structures and algorithms. Programs are written in the Java language.

CS 107 — Computer Architecture and Organization

(3) — CSU, UC

Prerequisites: CS 105

Skills Advisories: Eligibility for ENG 103

Hours: 90 (36 lecture, 54 lab)

The organization and behavior of real computer systems at the assembly-language level. The mapping of statements and constructs in a high-level language into sequences of machine instructions is studied, as well as the internal representation of simple data types, pointers, structures, and non-numeric data. Numerical computation is also examined.

CS 108 — Discrete Structures

(4) — CSU, UC

Prerequisites: MATH 150

Skills Advisories: Eligibility for ENG 103

Hours: 72 lecture

Introduction to the study of discrete objects, with a focus on applications in computer science. Topics include logic and proofs, sets, functions, sequences, sums, algorithms, integers, induction, recursion, counting, relations, graphs and trees.

CS 111 — HTML and Webmastering

(3) — CSU

Skills Advisories: Eligibility for ENG 103

Hours: 90 (36 lecture, 54 lab)

Project-oriented focus on the creation of web pages and the technology behind the web. Includes Hypertext Markup Language (HTML), Cascading Style Sheets (CSS), Forms, Extensible Markup Language (XML), Common Gateway Interface (CGI) and Content Management Systems (CMS). Students must also enroll in a CS 111 lab.

CS 114 — Python Programming

(3) — CSU, UC

Skills Advisories: Eligibility for ENG 103

Hours: 90 (36 lecture, 54 lab)

Introduction to Python language programming. Covers the python data model (objects, values and types), import system, expressions (unary, binary, comparisons, lambdas and expression lists), simple and compound statements, objects and object-oriented programming, and the Python library. Features problem-solving with Python across multiple domains, basic software engineering principles, and debugging techniques.

CS 115 — JavaScript and Dynamic HTML

(3) — CSU

Skills Advisories: Eligibility for ENG 103

Course Advisories: COMSC 111

Hours: 90 (36 lecture, 54 lab)

Project-oriented introduction to JavaScript programming, and using JavaScript with cascading style sheets to implement cutting-edge webpage effects with Dynamic HTML.

CS 116 — Web Server Programming

(3) — CSU

Skills Advisories: Eligibility for ENG 110 or 110H

Course Advisories: CS 111 or 120

Hours: 90 (36 lecture, 54 lab)

Project-oriented class that explores programming a Web server using PERL, Active Server Pages (ASP), Personal Home Pages (PHP) and Python, with an emphasis on PERL and PHP. Class develops Common Gateway Interface (CGI) scripts and Internet applications using these common tools. Includes such topics as e-commerce, security, browser independence and database integration.

CS 120 — Java Programming

(3) — CSU, UC

Skills Advisories: Eligibility for ENG 103

Course Advisories: CS 101 and/or CIS 101

and/or CS 200

Hours: 90 (36 lecture, 54 lab)

Study of the Java programming language. Topics include classes, encapsulation, inheritance, packages and methods. Students study, design, and develop multiple applications in the course.

CS 123 — Android Programming

(1.5) — CSU, UC*

Eligibility for English 110 or 110H

Course Advisories: CS 120 or 125

Hours: 42 (18 lecture, 24 lab)

Project-based development of applications for the Android platform. Students use state-of-the-art tools and frameworks to build and analyze programs that incorporate user interfaces, web services, animation, multimedia and location awareness. (*UC transfer limit: No credit for CS 122, 123 or 129 unless taken subsequent to or concurrently with CS 120*)

CS 125 — C# Programming

(1.5) — CSU

Skills Advisories: Eligibility for ENG 103

Course Advisories: CS 120 or 140

Hours: 86 (38 lecture, 48 lab)

Study of the programming language C# (C sharp). Definition of data types, loop control structures, functions, parameter passing, pointers, recursion, records, data structures, object-oriented techniques, the .NET framework, exception handling, interfaces, scoping rules, and supplied system objects.

CS 126 — Microsoft Mobile Device Programming

(1.5) — CSU

Skills Advisories: Eligibility for ENG 110 or 110H

Course Advisories: CS 120 or 125

Hours: 45 (18 lecture, 27 lab)

The use of C# in developing applications for wireless devices, such as mobile phones and personal digital assistants. Students use mobile device frameworks to create user interfaces and access data from databases, XML documents and web services.

CS 127 — ASP.NET Using C#

(1.5) — CSU

Course Advisories: CS 125

Hours: 45 (18 lecture, 27 lab)

Study of the use of C# in developing Active Server Page (ASP)-based dynamic websites. The use of the .NET framework, working with data and XML, error handling, ASP.NET server controls, custom controls and optimizing applications.

CS 129 — J2EE Server Programming

(1.5) — CSU, UC*

Skills Advisories: Eligibility for ENG 100

Course Advisories: CS 120

Hours: 45 (18 lecture, 27 lab)

Project-oriented introduction to Java 2 Enterprise Edition (J2EE) Web application development. Students learn to design, build and deploy Web applications. Servlets, Java Server Pages, Java DataBase Connectivity, JavaMail, eXtensible Markup Language processing and Enterprise JavaBeans are investigated. (**UC transfer limit: No credit for CS 122, 123 or 129 unless taken subsequent to or concurrently with CS 120*)

CS 130 — Introduction to the Linux Operating System**(2.5) — CSU, UC***Skills Advisories: Eligibility for ENG 103 and Proficiency in MATH 4**Course Advisories: CS 101**Hours: 63 (36 lecture, 27 lab)*

Survey of the Unix/Linux operating system and related subject matter. Topics include Unix/Linux architecture, commands, file system, processes and bash shell environment. Lectures and computer laboratory exercises provide a moderate-depth understanding of Unix/Linux architecture and commands from a computer science perspective.

CS 131 — Assembly Language Programming**(4) — CSU, UC***Skills Advisories: Eligibility for ENG 103**Course Advisories: CS 135**Hours: 108 (54 lecture, 54 lab)*

Introduction to basic computer organization, using Assembly language. Topics include computer hardware, machine language, data representation, binary manipulations, Boolean algebra, digital logic circuits, computer architecture and design. Assembly language programs developed on the college's computer.

CS 132 — Digital Logic Design**(3) — CSU, UC***Skills Advisories: Eligibility for ENG 103 and proficiency in MATH 104 or 107 or 111**Hours: 90 (36 lecture, 54 lab)*

Introduction to digital systems and logic design. Logic of propositions, boolean algebra, minterm and maxterm expansions, Karnaugh maps, Quine-McCluskey methods, multi-level circuits, combinational and sequential circuit design and timing diagrams, multiplexers, decoders, programmable logic devices, latches and flip-flops, finite state machines, registers and counters, register transfer language, simulation and debugging.

CS 133 — Introduction to Programming for Engineers**(3) — CSU, UC***Skills Advisories: Eligibility for ENG 110 or 110H; MATH 107**Course Advisories: CS 101 or CIS 101**Hours: 90 (36 lecture, 54 lab)*

General philosophy of programming for engineering majors. Students are introduced to a modern programming language (Matlab). Specific areas of study include algorithms, basic decision structures, arrays, matrices and graphing.

CS 135 — Programming Fundamentals**(3) — CSU, UC***Skills Advisories: Eligibility for ENG 103**Course Advisories: CS 101**Hours: 68-72 lecture*

Study of fundamental programming concepts. Topics include structured and OOP programming, definition of data types, nested IFs, looping techniques, CASE statements, procedures, functions, value and address parameters, file structures, dynamic list structures and recursion.

CS 137 — C Programming**(3) — CSU, UC***Skills Advisories: Eligibility for ENG 103**Course Advisories: CS 131 or 135**Hours: 78 (42 lecture, 36 lab)*

Study of the programming language C. Definition of data types, loop controls structures, functions, parameter passing, pointers, recursion, records data structures, object-oriented techniques and the Unix operating system.

CS 140 — Object-Oriented Programming Using C++**(4) — CSU, UC***Skills Advisories: Eligibility for ENG 103**Course Advisories: CS 120 or 137**Hours: 108 (54 lecture, 54 lab)*

Study of the object-oriented programming paradigm, including objects, messages, encapsulation, classes, inheritance and implementation issues. Implementations written in the object-oriented language C++.

CS 142 — Windows Programming with C++.NET**(3) — CSU***Skills Advisories: Eligibility for ENG 103**Course Advisories: CS 140**Hours: 90 (36 lecture, 54 lab)*

Study of Microsoft Windows programming with Visual C++.NET. Explores the difference between legacy windows programming using the Windows Software Developer's Kit (SDK) and Microsoft foundation classes (MFC) with state-of-the-art Windows.NET programming tools.

CS 143 — Discrete Math

(4) — CSU, UC

Prerequisites: MATH 150

Skills Advisories: Eligibility for ENG 103

Hours: 72 lecture

Introduction to the study of discrete objects, with a focus on applications in computer science. Topics include logic and proofs, sets, functions, sequences, sums, algorithms, integers, induction, recursion, counting, relations, graphs and trees.

CS 145J — Introduction to Data Structures

(3) — CSU, UC

Skills Advisories: Eligibility for ENG 103

Course Advisories: CS 120 or 135

Hours: 78 (42 lecture, 36 lab)

Study of data structures and algorithms. Design, coding and testing of linked lists, trees, queues, stacks, hash tables, and other dynamic data structures, as well as searching and sorting algorithms. Time and space analysis of data structures and algorithms. Programs are written in the Java language.

CS 145P — Introduction to Data Structures

(3) — CSU, UC

Skills Advisories: Eligibility for ENG 103

Course Advisories: CS 135

Hours: 78 (42 lecture, 36 lab)

Study of data structures and algorithms. Design, coding and testing of linked lists, trees, queues, stacks, hash tables, and other dynamic data structures, as well as searching and sorting algorithms. Time and space analysis of data structures and algorithms. Programs are written in the Pascal language.

CS 165 — Software Design Patterns

(1.5) — CSU

Skills Advisories: Eligibility for ENG 110 or 110H

Course Advisories: CS 120 or 140

Hours: 45 (18 lecture, 27 lab)

Introduction to software design patterns and their use in object-oriented systems. Creational, structural and behavioral patterns are investigated. Real world examples by acknowledged experts are studied. Students strengthen their software design skills by applying patterns in course projects. State of the art development tools are used throughout the course.

CS 180 — Software Engineering with UML

(3) — CSU, UC

Skills Advisories: Eligibility for ENG 110 or 110H

Course Advisories: CS 120

Hours: 90 (36 lecture, 54 lab)

Study of software engineering and component-based design, using the Unified Modeling Language (UML). Students employ a standard software engineering process that includes requirements analysis, design, implementation and testing. Students learn about various UML diagrams and use them to express software requirements and designs. Investigates rapid application development, using state-of-the-art tools and component libraries.

CS 187 — iOS Programming

(3) — CSU

Skills Advisories: Eligibility for ENG 110 or 110H

Course Advisories: CS 135

Hours: 90 (36 lecture, 54 lab)

Project-oriented course in developing Objective-C based native applications for the iOS devices (e.g. iPhone, iPad). Covers development tools (Xcode, Interface Builder, Instruments). Objective-C language, Cocoa and Cocoa Touch frameworks, as well as deployment to the App store.

CS 189 — Programming Practicum

(1) — CSU

Hours: 18 lecture

Programming course focused on rapid team-based, problem-solving techniques. Problems are rapidly classified based on difficulty and the ability to partition the problem across the team. Algorithms are developed to solve those problems using standard tools and libraries. Emphasis on leadership skills, team dynamics and team problem-solving.

CS 200 — Introduction to Programming

(4) — CSU, UC

Skills Advisories: Eligibility for ENG 100 and 103

Hours: 72 lecture

Exposes students to the fundamental concepts of programming using an object-oriented language and is intended as an introductory level programming course. Ideal for beginners and also serves as the first course in a sequence that is compliant with the standards of the Association for Computing Machinery.