

COMPUTER SCIENCE

CS V04 - INTRODUCTION TO COMPUTERS AND COMPUTER LITERACY - 3 Units

Hours: 2.5 lecture, 1.5 laboratory weekly

This course introduces the student to the fundamental concepts of computers and computer literacy. Topics include information technology, operating systems, applications software, hardware components, input, output, storage, graphics, hypermedia, multimedia, networking, computer communications, the Internet, Web page features, computer security, privacy with computer communications, systems analysis, systems design, programming languages, programming methods, management information systems, and the ACM Code of Ethics and Professional Conduct. This course also offers the student an introduction to the historical and social context of computing, including an overview of computer and information sciences as a discipline.

Fees will be required. Field trips may be required. Transfer credit: CSU; UC; credit limitations - see counselor. **CAN CSCI 2.**

CS V11 - PROGRAMMING FUNDAMENTALS - 3 Units

Recommended preparation: CS V04 or equivalent; and MATH V03 or 1 year of high school intermediate algebra (Algebra II) with grade of C or better

Hours: 2.5 lecture, 1.5 laboratory weekly

This course introduces the student to fundamental concepts of procedural programming. Topics include data types, control structures, functions, arrays, files, and the mechanics of running, testing, and debugging. The course emphasizes good software engineering principles and developing fundamental programming skills in the context of a functional programming language.

Fees will be required. Field trips may be required. Transfer credit: CSU; UC. **CAN CSCI 22.**

CS V13 - OBJECT-ORIENTED PROGRAMMING - 3 Units

Prerequisite: CS V11 or equivalent; and MATH V03 or 1 year of high school intermediate algebra (Algebra II) with grade of C or better

Hours: 2.5 lecture, 1.5 laboratory weekly

This course introduces the concepts of object-oriented programming to students with a background in the procedural paradigm. The course begins with a review of control structures and data types with emphasis on structured data types and array processing. It then moves on to introduce the object-oriented programming paradigm, focusing on the definition and use of classes along with the fundamentals of object-oriented design. Other topics include an overview of programming language principles, simple analysis of algorithms, basic searching and sorting techniques, and an introduction to software engineering issues.

Fees will be required. Field trips may be required. Transfer credit: CSU; UC.

CS V15 - DATA STRUCTURES AND ALGORITHMS - 3 Units

Prerequisite: CS V13 or equivalent; and MATH V20 or both MATH V04 and MATH V05

Hours: 2.5 lecture, 1.5 laboratory weekly

This course builds on the foundation provided by the programming fundamentals/object-oriented programming sequence to introduce the fundamental concepts of data structures and the algorithms that proceed from them. Topics include recursion, the underlying philosophy of object-oriented programming, fundamental data structures (including stacks, queues, linked lists, hash tables, trees, and graphs), the basics of algorithmic analysis, and an introduction to the principles of language translation.

Fees will be required. Field trips may be required. Transfer credit: CSU; UC. **CAN CSCI 24.**

CS V17 - DISCRETE STRUCTURES - 3 Units

Prerequisite: MATH V20 or both MATH V04 and MATH V05

Recommended preparation: CS V11 or equivalent

Hours: 3 lecture weekly

This course introduces the student to discrete mathematics as it is used in computer science. Topics include functions, relations, sets, propositional and predicate logic, simple circuit logic, proof techniques, elementary combinatorics, and discrete probability.

Field trips may be required. Same as MATH V52. Transfer credit: CSU; UC; credit limitations - see counselor. **CAN CSCI 26.**

CS V19 - COMPUTER ARCHITECTURE AND ORGANIZATION - 3 Units

Prerequisite: CS V15 or equivalent; and CS V17 or equivalent

Hours: 2.5 lecture, 1.5 laboratory weekly

This course introduces students to the organization and architecture of computer systems, beginning with the standard von Neumann model and then moving forward to more recent architectural concepts. This course also offers the students an introduction to assembly language for low-level programming of system software and computer applications.

Fees will be required. Field trips may be required. Transfer credit: CSU; UC. **CAN CSCI 10.**

CS V20 - BEGINNING VISUAL BASIC - 3 Units

Prerequisite: CS V04 or equivalent; and MATH V03 or 1 year of high school intermediate algebra (Algebra II) with grade of C or better

Recommended preparation: CS V11 or equivalent

Hours: 2.5 lecture, 1.5 laboratory weekly

This course introduces the student to the concepts of event-driven programming. This course presents the knowledge, skills and techniques to build Visual Basic programs. Topics include the programming design process, visual console procedures, an overview of computer hardware organization, data representation, instruction forms, flowcharting problems, logic solutions, and algorithms for problem solution. Concepts and definitions include: labels, variables, subscripts, data structures, object-oriented design, OLE, API, Windows forms, and scientific notation. Also included are debugging Visual Basic statements and elementary coding.

Fees will be required. Field trips may be required. Transfer credit: CSU; UC. **CAN CSCI 6.**

CS V22 - INTERMEDIATE VISUAL BASIC - 3 Units

Prerequisite: CS V11 or CS V20 or equivalent

Recommended preparation: MATH V03 or equivalent

Hours: 2.5 lecture, 1.5 laboratory weekly

This intermediate-level course presents to students the concepts of developing Microsoft Windows applications with Visual Basic. Topics include Multiple Document Interface (MDI), graphics, custom controls Dynamic Link Libraries (DLLs), Object Linking and Embedding (OLE), advanced database access, and the Windows API. Emphasis is placed on developing more advanced programming techniques using programming standards, application integration, and rapid development.

Fees will be required. Field trips may be required. Transfer credit: CSU; UC.

CS V24 - ADVANCED VISUAL BASIC - 3 Units

Prerequisite: CS V22 or equivalent

Hours: 2.5 lecture, 1.5 laboratory weekly

This course presents object-oriented programming with Visual Basic to the student. The course begins with a review of advanced-level event model design methods for Single-Document Interface (SDI) and Multiple-Document Interface (MDI) applications. It then moves on to introduce the object-oriented programming paradigm, focusing on the definition and use of classes along with the fundamentals of object-oriented design. Other topics include an overview of programming language principles, analysis of algorithms, building ActiveX components and an introduction to software engineering issues.

Fees will be required. Field trips may be required. Transfer credit: CSU; UC.

CS V30 - BEGINNING C++ - 3 Units

Prerequisite: CS V04 or equivalent; and MATH V03 or 1 year of high school intermediate algebra (Algebra II) with grade of C or better

Hours: 2.5 lecture, 1.5 laboratory weekly

This course introduces the student to the concepts of object-oriented programming. The course begins with a review of control structures and data types with emphasis on structured data types and array processing. It then moves to introduce the object-oriented program paradigm, focusing on the definition and use of classes along with fundamentals of object-oriented design. Other topics include an overview of programming language principles, analysis of algorithms, and an introduction to software engineering issues. A complete object-oriented development framework is presented that encourages extensibility, reusability, and manages complexity.

Fees will be required. Field trips may be required. Transfer credit: CSU; UC. **CAN CSCI 18.**

CS V32 - INTERMEDIATE C++ - 3 Units

Prerequisite: CS V13 or CS V30 or equivalent

Hours: 2.5 lecture, 1.5 laboratory weekly

This course will cover intermediate C++ programming using the Windows environment. The Windows-based programs created will incorporate modern object-oriented design methods. Advanced programming concepts, design methods, and implementation of C++ in an object-oriented Windows-based environment are included. Students will create sophisticated applications utilizing a graphical user interface.

Fees will be required. Field trips may be required. Transfer credit: CSU; UC.

CS V34 - ADVANCED C++ - 3 Units

Prerequisite: CS V32 or equivalent

Hours: 2.5 lecture, 1.5 laboratory weekly

This course presents to the student advanced C++ and the implementation of enterprise level software architectures. Topics include enhancing maintainability of software with error handling, pointers in a Common Object Model (COM+) environment, string details in C++ for international development, COM+ currency, using COM+ objects in interoperable development environments, Active Template Library (ATL) with multiple inheritance, cross-platform data-transformation services, marshal-by-value implementations in C++, COM+ reference-cycle management, and the Standard Template Library (STL) in the COM+ realm. This course also covers: software systems with a COM+ four-tier business-object application pattern; designing solutions with a Simple Object Access Protocol (SOAP); the Microsoft Management Queue (MSMQ), designing scalable systems with Microsoft Transaction Server (MTS); and understanding data access in the COM+ age.

Fees will be required. Field Trips may be required. Transfer credit: CSU; UC.

CS V40 - BEGINNING JAVA - 3 Units

Prerequisite: CS V04 or equivalent; and MATH V03 or 1 year of high school intermediate algebra (Algebra II) with grade of C or better

Hours: 2.5 lecture, 1.5 laboratory weekly

This course introduces the student to fundamental concepts of object-oriented programming with Java. Topics include data types, control structures, functions, arrays, files, and the mechanics of running, testing, and debugging. It then moves on to introduce the object-oriented programming paradigm, focusing on the definition and use of classes along with the fundamentals of object-oriented design. The course also offers an introduction to the historical and social context of computing and an overview of computer science as a discipline.

Fees will be required. Field trips may be required. Transfer credit: CSU; UC.

CS V42 - INTERMEDIATE JAVA - 3 Units

Prerequisite: CS V13 or CS V40 or equivalent

Hours: 2.5 lecture, 1.5 laboratory weekly

This course builds on the student's knowledge of object-oriented design and provides the student with the skills and techniques to create Java application and applets. Topics included in this course are the Java language, Java API, Java programming techniques, integrating graphics, security issues, Java tools, Java applets, and JavaScript.

Fees will be required. Field trips may be required. Transfer credit: CSU; UC.

CS V44 - ADVANCED JAVA - 3 Units

Prerequisite: CS V42 or equivalent

Hours: 2.5 lecture, 1.5 laboratory weekly

This course presents to the student the skills and knowledge needed to create advanced Java applications. Included in this course are advanced Java applications design theory, advanced Swing features, database manipulations, advanced thread usage, Java Server Pages with JDBC, distributed computing between Java programs, streams-based sockets, object serialization, advanced reusable software components and advanced security issues.

Fees will be required. Field trips may be required. Transfer credit: CSU; UC.

CS V80 - BEGINNING C - 3 Units

Prerequisite: CS V04 or equivalent; and MATH V03 or 1 year of high school intermediate algebra (Algebra II) with grade of C or better

Recommended preparation: CS V11 or equivalent

Hours: 2.5 lecture, 1.5 laboratory weekly

This course introduces the students to programming with C language. Included in this course are program design process, C operators, standard I/O functions, strings, pointers, arrays, datatypes, storage classes, file operations, and design of common algorithms. Algorithms are implemented as structured programs in C language.

Fees will be required. Field trips may be required. Transfer credit: CSU; UC. **CAN CSCI 16.**

CS V82 - UNIX SYSTEMS PROGRAMMING - 3 Units

Prerequisite: CS V80 or equivalent

Hours: 2.5 lecture, 1.5 laboratory weekly

This course builds on the student's knowledge of the C programming language and introduces UNIX features and facilities. The course helps the student to explore issues related to programming in a UNIX environment. The difference between programming in a DOS/Windows environment and in a UNIX/X-Windows environment is covered focusing on the UNIX system call interface, the programming interface between the UNIX Kernel and applications software running in the UNIX environment. Students will create 32-bit applications in the UNIX programming environment.

Fees will be required. Field trips may be required. Transfer credit: CSU; UC.

CS V86 - FORTRAN PROGRAMMING - 3 Units

Prerequisite: CS V04 or equivalent; and MATH V03 or 1 year of high school intermediate algebra (Algebra II) with grade of C or better
Recommended preparation: CS V11 or equivalent
Hours: 2.5 lecture, 1.5 laboratory weekly

This course presents to the student the concept of programming with FORTRAN. Topics include numerical computing, data types, control structures, programs and subprograms, dynamic memory allocation, pointers, arrays, files, and the mechanics of running, testing and debugging. The course emphasizes good software engineering principles and developing fundamental programming skills in the context of the FORTRAN programming language.

Fees will be required. Field trips may be required. Transfer credit: CSU; UC. **CAN CSCI 4.**

CS V88 - COMPUTER SCIENCE WORKSHOPS - .5-10 Units

Prerequisite: varies with topic
Hours: lecture and/or laboratory as required by unit formula

Designed to meet specific needs of the college and community, as required and requested by persons whose needs in this area are not met by present course offerings.

Fees may be required. Courses with same title may not be repeated; may be taken for a maximum of 4 times.

CS V89 - WORKSHOPS IN COMPUTER SCIENCE - .5-10 Units

Prerequisite: varies with topic
Hours: lecture and/or laboratory as required by unit formula

Designed to meet specific needs of the college and community, as required and requested by persons whose needs in this area are not met by present course offerings.

Fees may be required. Courses with same title may not be repeated; may be taken for a maximum of 4 times. Transfer credit: CSU; for UC, determined after admission.

CS V90 - DIRECTED STUDIES IN COMPUTER SCIENCE - 1-6 Units

Prerequisite: varies with topic
Hours: lecture and/or laboratory as required by unit formula

This course offers specialized study opportunities for students with intermediate skills, who wish to pursue projects not included in the regular curriculum. Students are accepted only by written project proposal approved by the discipline prior to enrollment.

May be taken for a maximum of 4 times not to exceed 6 units. Transfer credit: CSU; for UC, determined after admission.

CS V95 - COMPUTER SCIENCE INTERNSHIP I - 1-4 Units

Corequisite: enrolled in minimum 7 units to include internship
Recommended preparation: completion of or concurrent enrollment in one course in the discipline
Hours: 60 per unit

This computer science internship course offers students with intermediate skills who are volunteers (unpaid) an opportunity to obtain work experience related to their field of study. Students are accepted as a result of consultation with a designated faculty member in the discipline and the acceptance of an approved work proposal.

Field trips will be required. May be taken for a maximum of 4 times, not to exceed 16 units total in combination with any other work experience/internship courses. Offered on a credit/no credit basis only. Transfer credit: for CSU, credit limitation—see counselor; for UC, determined after admission.

CS V96 - COMPUTER SCIENCE INTERNSHIP II - 1-4 Units

Corequisite: enrolled in minimum 7 units to include internship
Recommended preparation: completion of or concurrent enrollment in one course in the discipline
Hours: 75 per unit

This computer science internship course offers students with intermediate skills who are employed in the field an opportunity to expand their work experience related to their field of study. Students are accepted as a result of consultation with a designated faculty member in the discipline and the acceptance of an approved work proposal.

Field trips will be required. May be taken for a maximum of 4 times, not to exceed 16 units total in combination with any other work experience/internship courses. Offered on a credit/no credit basis only. Transfer credit: for CSU, credit limitation—see counselor; for UC, determined after admission.