

COMPUTER SCIENCE I

Computer Science I introduces the structured techniques necessary for efficient solution of business-related computer programming logic problems and coding solutions into a high-level language. The fundamental concepts of programming are provided through explanations and effects of commands and hands-on utilization of lab equipment to produce correct and accurate outputs. Topics include program flowcharting, pseudo coding, and hierarchy charts as a means of solving problems. The course covers creating file layouts, print charts, program narratives, user documentation and system flowcharts for business problems; algorithm development and review, flowcharting, input/output techniques, looping, modules, selection structures, file handling, and control breaks and offers students an opportunity to apply skills in a laboratory environment.

- DOE Code: 4801
- Recommended Grade Level: 10, 11, 12
- Required Prerequisites: Introduction to Computer Science or teacher confirmation of student demonstration of mastery of the Intro to Computer Science standards
- Credits: 2 semester course, 2 semesters required, 1 credit per semester, 2 credits maximum
- Counts as a Directed Elective or Elective for all diplomas

Dual Credit

This course provides the opportunity for dual credit for students who meet postsecondary requirements for earning dual credit and successfully complete the dual credit requirements of this course.

Application of Content and Multiple Hour Offerings

Intensive laboratory applications are a component of this course and may be either school based or work based or a combination of the two. Work-based learning experiences should be in a closely related industry setting. Instructors shall have a standards-based training plan for students participating in work-based learning experiences. When a course is offered for multiple hours per semester, the amount of laboratory application or work-based learning needs to be increased proportionally.

Career and Technical Student Organizations (CTSOs)

Career and Technical Student Organizations are considered a powerful instructional tool when integrated into Career and Technical Education programs. They enhance the knowledge and skills students learn in a course by allowing a student to participate in a unique program of career and leadership development. Students should be encouraged to participate in Business Professional of America, DECA, or Future Business Leaders of America, the CTSOs for this area.

Content Standards

Domain – Problem Analysis

Core Standard 1 Students analyze a problem and develop a solution by creating a computer program.

Standards

- CS1-1.1 Identify how to use a computer program to solve a problem
- CS1-1.2 Construct interactive computer programs that accept various forms of input and produce various forms of output, as a solution to a computer programming problem

- CS1-1.3 Use print charts, file layouts, program narratives, hierarchy charts, and system flowcharts, which accurately depict the problem assigned and describe the solution
- CS1-1.4 Report the program schematics and usage
- CS1-1.5 Identify the standard program flowchart symbols and use them correctly within the context of the basic control structures of sequence, selection and looping

Domain – Software Tools

Core Standard 2 Students apply and adapt software tools to develop a computer program.

Standards

- CS1-2.1 Construct a program that processes information
- CS1-2.2 Identify programming languages as procedural or object oriented
- CS1-2.3 Develop programs using reusable modules (modularization)
- CS1-2.4 Use debugging techniques to correct and validate the computer program
- CS1-2.5 Construct the program in a high-level programming language based on a created design
- CS1-2.6 Construct a program that opens and closes a file

Domain – Algorithm

Core Standard 3 Students design a solution to the problem using algorithms.

Standards

- CS1-3.1 Develop algorithms to solve a computer programming problem(s)
- CS1-3.2 Assess the use of algorithms to provide a solution to a programming problem
- CS1-3.3 Use pseudo code to describe a solution to a programming problem
- CS1-3.4 Create a program flowchart and ANSI standard flowcharting symbols to define a solution to a programming problem
- CS1-3.5 Explain how the algorithm can be used to solve a problem

Domain – Program Development

Core Standard 4 Students create a functional computer program.

Standards

- CS1-4.1 Define the process of programming.
- CS1-4.2 Create a computer program that corresponds to an algorithm or proposed solution
- CS1-4.3 Define programming structures
- CS1-4.4 Recognize data variables and constants
- CS1-4.5 Recognize local scope and global scope
- CS1-4.6 Use conditionals (IF statements)
- CS1-4.7 Use loops (while statements, for statements)
- CS1-4.8 Define single and multidimensional Arrays
- CS1-4.9 Use functions and methods to break down the program logic and support reuse
- CS1-4.10 Define the graphical user interface
- CS1-4.11 Identify the parts of the programming platform
- CS1-4.12 Identify different types of errors and handle them programmatically
- CS1-4.13 Use the order of operations when using calculations

Domain – Program Verification and Debugging

Core Standard 5 Students prove a computer program solution works by using verification and debugging techniques.

Standards

- CS1-5.1 Predict and explain output
- CS1-5.2 Identify cause/effect for input/output
- CS1-5.3 Perform input validation
- CS1-5.4 Scrutinize peers code for errors

Domain – Documentation

Core Standard 6 Students connect an associated task with the code by providing documentation.

- CS1-6.1 Describe the function of a computer program
- CS1-6.2 Identify the purposes of a computer program
- CS1-6.3 Explain concepts related to a computer program
- CS1-6.4 Describe how to use a computer program
- CS1-6.5 Identify cause/effect by explaining input and output
- CS1-6.6 Interpret input/output