GAVILAN COLLEGE

5055 Santa Teresa Blvd
Gilroy, CA 95023

Course Outline

COURSE: CSIS 45   DIVISION: 50   ALSO LISTED AS:

TERM EFFECTIVE: Fall 2021   CURRICULUM APPROVAL DATE: 12/14/2021

SHORT TITLE: C++ PROGRAMMING I

LONG TITLE: C++ Programming I

Units	Number of Weeks	Type	Contact Hours/Week	Total Contact Hours
3	18	Lecture: 2		36
		Lab: 3		54
		Other: 0		0
		Total: 5		90

Out of Class Hrs:	72.00
Total Learning Hrs:	162.00

COURSE DESCRIPTION:

An introduction to the concepts and methods of computer programming using C++. Students will be introduced to procedural and object-oriented programming design methodology. Topics covered include variable and constant declarations, selection statements, repetition, functions and recursion, arrays, strings, pointers, and an introduction to classes and objects. This course will prepare students for the Programming II class. This course has the option of a letter grade or pass/no pass. (C-ID: COMP 122) ADVISORY: CSIS 42

CREDIT STATUS: D - Credit - Degree Applicable

GRADING MODES
        L - Standard Letter Grade
        P - Pass/No Pass

REPEATABILITY: N - Course may not be repeated

SCHEDULE TYPES:

      02 - Lecture and/or discussion

      03 - Lecture/Laboratory

      04 - Laboratory/Studio/Activity

      04A - Laboratory - LEH 0.65

      05 - Hybrid

      71 - Dist. Ed Internet Simultaneous

      72 - Dist. Ed Internet Delayed

      73 - Dist. Ed Internet Delayed LAB

      73A - Dist. Ed Internet LAB-LEH 0.65

STUDENT LEARNING OUTCOMES:

By the end of this course, a student should:

1. Analyze and explain the behavior of simple programs involving the fundamental C++ programming constructs.

2. Modify and expand short programs that use standard conditional and iterative control structures and functions.

3. Design, implement, test, and debug a program that uses each of the following fundamental programming constructs: basic computation, simple I/O, standard conditional and iterative structures, and the definition of functions.

COURSE OBJECTIVES:

By the end of this course, a student should:

1. Discuss and explain how a C++ program utilizes them: variables, identifiers, assignments, input and output, and data types and expressions.

2. Explain and demonstrate how to write programs using I/O.

3. Discuss and demonstrate the ways that you can combine arrays, structures, and classes to form intricately structured types such as arrays of structures, arrays of classes, and classes with arrays as member variables.

4. Discuss C++ templates and demonstrate their application.

5. Discuss and apply the techniques of structured (functional) decomposition to break a program into smaller pieces.

6. Describe the mechanics of parameter passing.

7. Choose appropriate conditional and iteration constructs for a given programming task.

8. Describe and demonstrate different forms of binding, visibility, scoping, and lifetime management.

9. Summarize and use pseudocode or a programming language to implement, test, and debug algorithms for solving simple problems.

COURSE CONTENT:

Curriculum Approval Date: 12/14/2021

LECTURE CONTENT:

4 Hours

Lectures: Introduction to Computers and C++ Programming I - Computer Systems, Programming and Problem-Solving, Introduction to C++, and Testing and Debugging. C++ Basics - Variables and Assignments, Input and Output, Data Types and Expressions, Simple Flow of Control, and Program Style.

2 Hours

Lectures: More Flow of Control - Using Boolean Expressions, Multiway Branches, More About C++ Loop Statements, and Designing Loops.

4 Hours

Lectures: Procedural Abstraction and Functions That Return a Value - Top-Down Design, Predefined Functions, Programmer-Defined Functions, Procedural Abstraction, Scope and Local Variables, and Overloading Function Names. Functions for All Subtasks - 'void' Functions, Call-By-Reference Parameters, Using Procedural Abstraction, Testing and Debugging Functions, and General Debugging Techniques.

4 Hours

Lectures: I/O Streams as an Introduction to Objects and Classes - Steams and Basic File I/O, Tools for Stream I/O, and Character I/O. Arrays - Introduction to Arrays, Arrays in Functions, Programming with Arrays, and Multidimensional Arrays.

4 Hours

Lectures: Strings and Vectors - An Array Type for Strings, The Standard 'string' Class, and Vectors. Pointers and Dynamic Arrays.

4 Hours

Lectures: Defining Classes - Structures, Classes, Abstract Data Types, and Introduction to Inheritance. Friends, Overloaded Operators, and Arrays in Classes - Friend Functions, Overloading Operators, Arrays and Classes, and Classes and Dynamic Arrays.

4 Hours

Lectures: Separate Compilation and Namespaces. Pointers and Linked Lists - Nodes and Linked Lists and Stacks and Queues.

4 Hours

Lectures: Recursion - Recursive Functions for Tasks, Recursive Functions for Values, and Thinking Recursively. Inheritance - Inheritance Basics, Inheritance Details, and Polymorphism.

4 Hours

Lectures: Exception Handling - Exception-Handling Basics and Programming Techniques for Exception Handling. Templates - Templates for Algorithm Abstraction and Templates for Data Abstraction. Standard Template Library - Iterators, Containers, and Generic Algorithms.

2 Hours

Written Final.

COURSE CONTENT(CONTINUED):

LAB CONTENT:

6 Hours

Lab: Complete programming projects such as: entering the C++ program shown in the display and then modifying or changing the program as directed and writing a program that uses several types of variables.

3 Hours

Lab: Demonstrate the purpose of a Boolean expression in C++ programming. Perform the commonly used methods for terminating an 'input loop'. Complete programming problems utilizing the concepts covered in lecture.

6 Hours

Lab: Complete programming problems utilizing the concepts covered in the lecture, including demonstrating several of your own functions. Demonstrate how a driver program is used for debugging.

6 Hours

Lab: Utilize a 'manipulator'. Demonstrate how an array is used, including sorting and searching. Complete programming problems utilizing concepts covered in lecture.

6 Hours

Lab: Complete programming problems specific to the concepts covered in lecture, including demonstrating the use of strings and vectors as they relate to arrays.

6 Hours

Lab: Perform some techniques that will help define your classes based on modern programming practices. Perform techniques for defining operations on objects as nonmember functions. Complete programming problems specific to the material covered in lecture.

6 Hours

Lab: Use namespaces. Demonstrate the application of 'nodes' and 'linked list'. Perform programming problems. Complete programming problems specific to the concepts covered in lecture.

6 Hours

Lab: Write and use recursive functions. Complete programming problems specific to the material covered in lecture.

7 Hours

Lab: Write and perform programming. Complete programming problems specific to the concepts covered in the lecture.

2 Hours

Practical Final.

METHODS OF INSTRUCTION:

Lectures, Computer Demonstrations, Programming Projects, Case Studies

OUT OF CLASS ASSIGNMENTS:

Required Outside Hours 8

Assignment Description

Out of Class Assignments: Read Introduction to Computers and C++ Programming I chapter. Read chapter on C++ Basics. Complete HW problems.

Required Outside Hours 4

Assignment Description

Out of Class Assignments: Read chapter on More Flow of Control. Complete HW problems.

Required Outside Hours 8

Assignment Description

Out of Class Assignments: Read chapter on procedural abstraction and functions that return a value. Read functions for all subtasks chapter. Complete programming problems.

Required Outside Hours 8

Assignment Description

Out of Class Assignments: Read chapter on I/O Streams as an Introduction to Objects and Classes and chapter on Arrays. Complete programming problems.

Required Outside Hours 8

Assignment Description

Out of Class Assignments: Read chapters on Strings and Vectors and Pointers and Dynamic Arrays. Complete programming homework.

Required Outside Hours 8

Assignment Description

Out of Class Assignments: Read chapter on Defining Classes and read chapter on friends, overloaded operators, and arrays in classes. Complete HW programming problems.

Required Outside Hours 8

Assignment Description

Out of Class Assignments: Read chapters on Separate Compilation and Namespaces and Pointers and Linked Lists. Complete programming HW.

Required Outside Hours 8

Assignment Description

Out of Class Assignments: Read chapter on Recursion and chapter on Inheritance. Do programming HW.

Required Outside Hours 8

Assignment Description

Out of Class Assignments: Read chapters on Exception Handling, Templates, and Standard Template Library. Complete programming HW.

OUT OF CLASS ASSIGNMENTS(CONTINUED):
Required Outside Hours 4
Assignment Description
Study/prepare for written and practical finals.

METHODS OF EVALUATION:
Writing assignments
Evaluation Percent 5
Evaluation Description
Writing assignments: 5% - 20%
Written Homework

Problem-solving assignments
Evaluation Percent 60
Evaluation Description
Problem-solving demonstrations: 40% - 70%
Homework Problems,
Exams

Skill demonstrations
Evaluation Percent 20
Evaluation Description
Skill demonstrations: 20% - 50%
Class Performance,

Performance Exams
Objective examinations
Evaluation Percent 15
Evaluation Description
Objective examinations: 5% - 20%
Multiple Choice,
True/False,
Matching Items,
Completion

REPRESENTATIVE TEXTBOOKS:
Problem Solving with C++, 10th Edition, Savitch, Walter, Pearson, 2018.
ISBN: 978-0134448282
12th Grade Verified by: MS Word

ARTICULATION and CERTIFICATE INFORMATION

      Associate Degree:

            GAV E2, effective 200630

      CSU GE:

      IGETC:

      CSU TRANSFER:

            Transferable CSU, effective 200630

            Not Transferable

      UC TRANSFER:

            Transferable UC, effective 200630

            Not Transferable

SUPPLEMENTAL DATA:

Basic Skills: N

Classification: Y

Noncredit Category: Y

Cooperative Education:

Program Status: 1 Program Applicable

Special Class Status: N

CAN: CSCI18

CAN Sequence: XXXXXXXX

CSU Crosswalk Course Department: COMP

CSU Crosswalk Course Number: 122

Prior to College Level: Y

Non Credit Enhanced Funding: N

Funding Agency Code: Y

In-Service: N

Occupational Course: B

Maximum Hours:

Minimum Hours:

Course Control Number: CCC000564665

Sports/Physical Education Course: N

Taxonomy of Program: 070710