

## **CSIT 214**

### **C++ Programming**

4 Credits

## Community College of Baltimore County Common Course Outline

### Description

**CSIT 214 – C++ Programming:** provides an introduction to C++ programming by applying object-oriented techniques, problem solving, and algorithm design. Topics include data types, control structures, input/output, Boolean and arithmetic expressions with an emphasis on applications using arrays, vectors, pointers, functions, structures, and files.

**Pre-requisites:** CSIT 210 or consent of Program Director

### Overall Course Objectives

Upon completion of this course, students will be able to:

1. identify the steps in the software design process;
2. design algorithms to be translated into working solutions;
3. demonstrate the process of testing, debugging, and validating a solution;
4. implement modular structures using user-defined functions;
5. identify data types and variable naming conventions;
6. develop programs that perform correct calculations to solve problems;
7. demonstrate input and output data methods from the keyboard and files;
8. develop algorithms using primitive data types, operators, and expressions;
9. construct programs using control structures;
10. construct programs that use arrays, vectors, structures, and pointers;
11. write programs using object-oriented techniques such as classes, overloading, and inheritance;
12. identify techniques for formatting data; and
13. examine the impact of testing and validating solutions when developing applications.

### Major Topics

- I. Program development cycle
- II. Data types
- III. Boolean and arithmetic expressions
- IV. Input and output
- V. Formatting output
- VI. Evaluating expressions
- VII. Selection statements
  - a. Simple if statements

The Common Course Outline (CCO) determines the essential nature of each course.  
For more information, see your professor's syllabus.

- b. Nested if statements
- VIII. Repetition
  - a. While loops
  - b. Do loops
  - c. For loops
- IX. Functions
- X. Structures
- XI. Vectors
- XII. Files
- XIII. Pointers
- XIV. Arrays
  - a. 1-Dimensional
  - b. 2-Dimensional
  - c. Using arrays with functions
- XV. Object-oriented concepts
  - a. Classes
  - b. Overloading
  - c. Constructors
  - d. Inheritance
  - e. Polymorphism

### **Course Requirements**

Grading will be determined by the individual faculty member, but shall include the following, at minimum:

- Five comprehensive programming projects
- Two exams
- Comprehensive final exam or capstone programming project

Date Revised: 5/4/2021

The Common Course Outline (CCO) determines the essential nature of each course.  
For more information, see your professor's syllabus.