HOSTOS COMMYNITY COLLEGE
DEPARTMENT OF MATHEMATICS AND COMPUTER SCIENCE

COURSE: **CSC** 215 MODERN PROGRAMMING LANGUAGES

CREDIT HOURS: 3.0

EQUATED HOURS: 3.0

CLASS HOURS: 3.0

PRE/COREQUISITE: MAT 210
PRE/COREQUISITE: ENG 93/ESL 91/ESL 93
OR PREREQUISITE: CSC 205

RECOMMENDED TEXTS: C++ How to Program (10th Edition) By Paul Deitel & Harvey Deitel ISBN-13: 978-0-13-444823-7

DESCRIPTION: This course provides an introduction to problem solving methods and algorithm development through the study of the program, control structures, and data structures of the C++ programming language. The main aspects of the course include: the concepts of procedural and object-oriented programming, algorithm design, control structures in C++, functions and recursions, arrays, pointers, characters and strings, structured data, file operations, classes.

Suggested grading policy: A minimum of three quizzes, midterm/project, and comprehensive final exam

GRADES: A, A-, B+, B, B-, C+, C, D, I, F.

Student Learning Obectives

1) Students will be able to demonstrate knowledge of hardware and software components of computers, and the importance of Secure Coding standards and principles.
2) Student will demonstrate ability to formulate and write programs with variables of different data types, using secure coding protocols.
3) Students will demonstrate ability to work with input/output variables, characters and strings within contexts of programming
4) Students will demonstrate fluency formulating and writing programs with logical operators for decision making
5) Students will demonstrate ability formulating and writing programs with recursive loops

6) Students while demonstrate ability to write programs using functions, arrays and pointers

COURSE OUTLINE:

Week Topics

1 **I. Introduction to Computers and Programming**
1. Computer systems: hardware and software
2. C++ development environmennt, programming and debugging
3. Secure Coding protocols e.g. hiding data
4. Hands on (lab): Introducing the Dev-C++/Microsoft Visual C++ Platform

2 **II. Introduction to C++**
1. The parts of a C++ program
2. The cout object, variables, constants, the assignment statement, identifiers, comments
3. Integer data types, floating-point data types
4. Arithmetic operators
5. char data type, the string class
6. The bool data type

3 **III. Expressions and Interactivity**
1. The `cin` object
2. Mathematical expressions
3. Multiple and combined assignment
4. Formatting output
5. Working with characters and string objects
6.More mathematical library functions
7. Introduction to file input and output

4 **IV. Making Decisions**
1. Relational operators
2. `if, if/else, if/else if` statements; nested `if` statements; menus
3. Logical operators
4. The conditional operator; the `switch` statement

5 **V. Looping**
1. Increment and decrement, the `while` loop, counters
2. The `do-while` loop
3. The for loop
4. Nested loops
5. Breaking out of a loop; the continue statement

6 **VI. Functions**

1. Introduction to recursions
2. Defining and calling functions; function prototypes
3. Sending data into a function; passing data by value; the return statement 4. Returning value from a function**; r**eturning a Boolean value
5. Local and global variables; static local variables; default arguments; the `exit()` function

7 **VII. Arrays**

1. Arrays hold multiple values; accessing array elements; no bounds checking in C++
2. Array initialization; processing array contents
3. Arrays as function arguments
4. 2-dimensional arrays; arrays of strings; arrays with 3 or more dimensions

8 **VIII. Pointers**

1. Getting the address of a variable; pointer variables; the relationship between arrays and pointers
2. Pointer arithmetic; initializing pointers; comparing pointers
3. Pointers as function parameters

9 **IX. Characters, Strings, the `string` Class**

10 **X. Structured Data**

1. Abstract data types; accessing structure members; initializing a structure
2. Arrays of structures; nested structures
3. Structures as function arguments; returning a structure from a function
4. Pointers to structures

11 **XI. Advanced File Operations**

12 **XII. Introduction to Classes**

1. Procedural and object-oriented programming
2. Introduction to classes
3. Defining an instance of a class; inline member functions;

13 4. Constructors; passing arguments to constructors; destructors;
5. Private member functions
6. Arrays of objects

14 Review