Course Synopsis: CSC 104 – Computer Programming II

Lagos State University  
Department of Computer Science  
100 Level – Second Semester  
Credit Units: 3

# Course Description / Synopsis:

This course is a continuation of CSC 103 and aims to deepen students’ understanding of programming concepts. It introduces more advanced programming topics including arrays, pointers (if applicable), structures, and functions. Students will also explore basic data structures such as stacks and queues, and concepts of recursion, file handling, and error management. The course emphasizes writing efficient, modular, and reusable code using a high-level programming language (typically Python or C).

# Course Objectives:

* Develop more complex programs using appropriate data structures.
* Understand and apply modular programming techniques using functions.
* Use arrays and strings effectively in solving computational problems.
* Implement recursion for suitable problems.
* Perform basic file input/output operations.
* Understand and implement simple data structures such as stacks and queues.
* Debug, test, and document code effectively.

# Course Outline / Topics:

1. Review of Basic Programming Concepts

* Data types, variables, operators, control structures

1. Functions and Modular Programming

* Function definition and calls
* Parameter passing (by value and by reference)
* Scope and lifetime of variables

1. Arrays and Strings

* One-dimensional and multi-dimensional arrays
* String handling and manipulation

1. Pointers and Memory Management (If applicable)

* Pointer basics
* Dynamic memory allocation

1. Structures and User-defined Data Types

* Defining and using structures
* Nested structures

1. Recursion

* Recursive functions
* Examples: factorial, Fibonacci series, towers of Hanoi

1. Introduction to Data Structures

* Stacks and Queues
* Basic operations (push, pop, enqueue, dequeue)

1. File Handling

* File operations: open, read, write, close
* Text vs. binary files

1. Error Handling and Debugging

* Types of errors (syntax, runtime, logical)
* Debugging techniques

# Teaching Methods:

* Interactive coding sessions
* Laboratory programming exercises
* Group projects
* Code reviews and debugging workshops

# Assessment Methods:

Continuous Assessment: 30%  
- Quizzes  
- Assignments  
- Practical labs

Final Examination: 70%