Resource Person: Afifa Wajid (afifa.wajid@umt.edu.pk)

Semester: Fall-2018

Course Title: Introduction to Computer Programming

Course Code: XI-513

Course Type: Foundation

Pre-Requisite: N/A

Counseling Hours: Weekdays (04:00PM to 6:00PM)

Program: MCS

Program Head: Mr. Imran Saleem

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Name** | **Signature** | **Date** |
| **Prepared By**(Resource Person) | Afifa Wajid |  |  |
| **Checked By**(Program Head) | Imran Saleem |  |  |
| **Approved By**(Director SPA) | Dr. Naveed Yazdani |  |  |

**Course Description**

This course is an introduction to computer programming using the C++ programming language. The course uses the C++ input/output system rather than the traditional C I/O. This course covers basic procedural techniques such as variables, data types, selection, iteration, and functions. It will also introduce students to arrays, application of arrays such as sorting and searching, and gives an overview of pointers and file handling. We'll also look at various general programming concepts, such as algorithm design and debugging. By the end of the course, students should be able to construct a moderately-sized program.

**Format of the Course:**

Weekly readings with supplement lecture material and Lab exercises.

**Instructional Goals (or Course Objective)**

* Be familiar with Integrated Development Environment of C
* Be familiar programming terminology such as variables, data types, selection, iteration, and functions
* Be able to write moderate size programs.
* Be able to use programming constructs to provide various solutions
* Be able to use debugging tool for removing errors and developing syntactically and semantically correct programs
* Be able to use programming conventions for program enhancement, maintenance and readability.

**Course (Student) Objectives**

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

* Demonstrate the ability to use different data types, including one-dimensional arrays, in computer programs
* Demonstrate the ability to use control structures and simple algorithms in computer programs;
* Demonstrate the ability to use subprograms within computer programs;
* Use proper code maintenance techniques and conventions when creating computer programs.
* Use a variety of problem-solving strategies to solve different types of problems
* Design algorithms per specifications;
* Use Sorting algorithms to sort arrays
* Relate the specifications of computer components to user requirements

**Brief Course Content**

**Session No. 1: Introduction**

* Course Introduction, Teaching & Assessment Methodology
* Distribution of Course Outlines
* Discussion on Course Outline
* What is Computer and computer programming language?
* Types of Programming Language,
* Interpreter, Compiler, Debugger, IDE
* Introduction to Algorithms
* Flowcharts
* Pseudo code

**Learning Objectives**

Participants will be exposed to different techniques to build logic for writing programs in C such as flowcharts and Pseudo code. This will help them in understanding the mechanism that how program will work and execute.

**Session 2: Introduction to C++ and IDE**

* Starting C++
* Our First Program
* Variables
* Data Types
* Arithmetic Operators
* Precedence of Operators
* Use of Operators

**Learning Objectives**

Participants will write their first program in C using IDE that will help them understand various aspects of IDE. They will be exposed to the concepts of memory using variables and data types by developing small programs.

**Activities**

* 1st Assignment
* Lab Task

**Session 3: Decision and Control Constructs**

* Decisions
* Conditional Statements
* if/else structure
* Logical Operators
* Repetition Structures
* Overflow Condition
* Infinite Loop
* Properties of While loop
* Do-While Statement
* for Statement
* Increment/decrement Operators
* Switch Statement
* Break Statement
* Continue Statement

**Learning Objectives**

Participants will learn various decision constructs such as IF/else, for, while to implement decision making and iteration to repeat certain code.

**Activities**

* 1st Quiz
* Lab Task

**Session 4: Arrays**

* Introduction of Arrays
* Initialization of Arrays
* Copying Arrays
* Linear Search
* Character Arrays
* Initialization of Character Arrays
* Arrays Comparison
* Sorting Arrays
* Searching arrays

**Learning objectives**

Participants will learn how to create one and two dimensional arrays, why they are used? They will learn the concepts of searching and sorting using arrays.

**Activities**

* 2nd Assignment
* Lab Task

**Session 5: Arrays and Functions**

* Multidimensional Arrays
* Array Manipulation
* Introduction to Functions
* Structure of a Function
* Function definition and Function prototype
* Function header and Function body
* Functions And arrays

**Learning Objectives**

Participants will learn the multidimensional arrays and the importance of function in programming. They will learn how to write and use simple function to improve programming solutions.

**Activities**

* 2nd Quiz

**Session 6: Use of Functions**

* Header Files
* Scope of Identifiers
* Function Call
* Call by Value

**Learning objectives**

Participants will learn how functions can be called differently and the impact of function on the outcome if called by value of reference. They will also learn how to combine related functions to be used as a header file.

**Activities**

* 3rd Quiz
* Lab task

**Session 7: Functions Continued**

* Call by Reference
* Call by Address
* Recursive Functions

**Learning objectives**

Participants will learn how functions can be called other than pass by value and they will also learn about recursive functions and different examples of recursive functions.

**Activities**

* Revision Session

**Session 8: MIDTERM**

**Session 9: Pointers**

* Introduction to Pointers
* Declaration of Pointers
* Pointers and Call by Reference
* Pointers, Strings and Arrays
* Relationship between Pointers and Arrays
* Pointer Expressions and Arithmetic

**Learning objectives**

Participants will learn a unique feature of pointers that may replace arrays in different programming scenarios. They will learn how to declare and use pointers in variety of problems. Participants will also learn different pointer operations that may be help full in increasing the effectiveness of the programs.

**Activities**

* 3rd Assignment
* Lab Task

**Session 10: Pointer Operations and String Operations**

* Multi-dimensional Arrays
* Pointers to Pointers
* String Manipulation Functions
* String Handling

**Learning objectives**

Participants will learn basic pointer and string operations and how to manipulate characters using arrays, pointers and functions.

**Activities:**

* 4th Assignment
* 4th Quiz

**Session 11: String Operations**

* String Manipulation Functions
* Character Handling Functions
* String Conversion Functions
* String Functions
* Search Functions

**Learning objectives**

Participants will learn the how to manipulate and perform the conversion and searching operations on the strings.

**Activities**

* 5th Quiz
* Lab Task

**Session 12: Structures**

* Introduction to Structures
* Declaration of a Structure
* Initializing Structures
* Functions and structures
* Arrays of structures

**Learning objectives**

Participants will learn the how to develop user defined data types using structures, how structures are used in variety of ways in solving different programming problems. They will also learn how to implement basic Object Oriented programming concepts using structures.

**Activities:**

* 5th Assignment

**Session 13: File Handling**

* Introduction to File Handling
* Text File Handling
* Output file handling
* Sequential and Random Access Files
* Position in a File

**Learning Objective:**

Participants will learn different file operations such as creation of files, data entry to file.

**Activities:**

* Lab Task

 **Session 14: File Handling**

* Setting the Position
* Efficient Way of Reading and Writing Files
* Copying a File in Reverse Order

**Learning Objective:**

Participants will learn modification of files and deletion of data and files. They will also learn how to develop a database using files

**Activities:**

* Revision of the course

**Session 15: Project**

* Project Viva’s & Presentations

**Session 16: FINALTERM**

**Recommended Book (s) & Text:**

* Deitel&Deitel, *C - How to Program 8th Edition*, Pearson – Prentice Hall

**E-Resources:**

 <http://lms.umt.edu.pk/course/view.php?id=8495>

**ASSESSMENT METHODOLOGY**

|  |  |
| --- | --- |
| Assignments / Lab Activities | 20% |
| Quizzes  | 15% |
| Project | 15% |
| Midterm | 15% |
| Final Term Exam | 35% |
| Total | 100% |

* Assignments can be either take-home or in class lab activities
* Quizzes mentioned in the outline are not definite, there can be surprise quizzes as well, be prepared.
* Semester project groups will have 2-3 members and each member has to justify their participation in the project to secure marks

**CALENDAR OF ACTIVITIES**

|  |  |  |  |
| --- | --- | --- | --- |
| **Session** | **Topic** | **Readings**  | **Activities** |
| 1 | **Introduction*** Course Introduction, Teaching & Assessment Methodology
* Distribution of Course Outlines
* Discussion on Course Outline
* What is Computer and computer programming language?
* Types of Programming Language,
* Interpreter, Compiler, Debugger, IDE
* Introduction to Algorithms
* Flowcharts
* Pseudo code
 | Lecture slidesCh#1, Intro to computers |  |
| 2 | **Introduction to C++ and IDE*** Starting C++
* Our First Program
* Variables
* Data Types
* Arithmetic Operators
* Precedence of Operators
* Use of Operators
 | Lecture slidesCh#2, Intro to C++ Programming | 1st AssignmentLab Task |
| 3 | **Decision and Control Constructs*** Decisions
* Conditional Statements
* if/else structure
* Logical Operators
* Repetition Structures
* Overflow Condition
* Infinite Loop
* Properties of While loop
* Do-While Statement
* for Statement
* Increment/decrement Operators
* Switch Statement
* Break Statement
* Continue Statement
 | Lecture slidesCh#4, Control Statements | 1st QuizLab Task |
| 4 | **Arrays*** Introduction of Arrays
* Initialization of Arrays
* Copying Arrays
* Linear Search
* Character Arrays
* Initialization of Character Arrays
* Arrays Comparison
* Sorting Arrays
* Searching arrays
 | Lecture slidesCh#7 Arrays  | 2nd AssignmentLab Task |
| 5 | **Arrays and Functions** * Multidimensional Arrays
* Array Manipulation
* Introduction to Functions
* Structure of a Function
* Function definition and Function prototype
* Function header and Function body
* Functions And arrays
 | Lecture slidesCh#6 & 7 Arrays and Intro to functions | 2nd Quiz  |
| 6 | **Use of Functions*** Header Files
* Scope of Identifiers
* Function Call
* Call by Value
 | Lecture slidesCh#6 Functions Implementations  | 3rd QuizLab task |
| 7 | **Functions Continued*** Call by Reference
* Call by Address
* Recursive Functions
 | Lecture slidesCh#6 Functions Implementations | Revision Session  |
| 8 | **MIDTERM** |
| 9 | **Pointers*** Introduction to Pointers
* Declaration of Pointers
* Pointers and Call by Reference
* Pointers, Strings and Arrays
* Relationship between Pointers and Arrays
* Pointer Expressions and Arithmetic
 | Lecture slidesCh#8, Pointers  | 3rd AssignmentLab Task |
| 10 | **Pointer Operations and String Operations*** Multi-dimensional Arrays
* Pointers to Pointers
* String Manipulation Functions
* String Handling
 | Lecture slidesCh#8, Pointers and Pointer based strings | 4th Assignment4th Quiz  |
| 11 | **String Operations** * String Manipulation Functions
* Character Handling Functions
* String Conversion Functions
* String Functions
* Search Functions
 | Lecture slidesOperations on Strings  | 5th Quiz Lab Task |
| 12 | **Structures** * Introduction to Structures
* Declaration of a Structure
* Initializing Structures
* Functions and structures
* Arrays of structures
 | Lecture slides | 5th Assignment |
| 13 | **File Handling*** Introduction to File Handling
* Text File Handling
* Output file handling
* Sequential and Random Access Files
* Position in a File
 | Lecture slidesCh#17 File Processing | Lab Task  |
| 14 | **File Handling*** Setting the Position
* Efficient Way of Reading and Writing Files
* Copying a File in Reverse Order
 | Lecture slides | Revision of the course  |
| 15 | * Project Viva’s & Presentations
 |  | Viva’s & Presentations |
| 16 | **FINALTERM** |