

**ALLAMA IQBAL OPEN UNIVERSITY, ISLAMABAD**  
(Department of Computer Science)

**WARNING**

1. **PLAGIARISM OR HIRING OF GHOST WRITER(S) FOR SOLVING THE ASSIGNMENT(S) WILL DEBAR THE STUDENT FROM AWARD OF DEGREE/CERTIFICATE, IF FOUND AT ANY STAGE.**
2. **SUBMITTING ASSIGNMENTS BORROWED OR STOLEN FROM OTHER(S) AS ONE'S OWN WILL BE PENALIZED AS DEFINED IN "AIOU PLAGIARISM POLICY".**

Course: Programming Language-I (3407)  
Level: BS (CS)

Semester: Autumn, 2013  
Total Marks: 100  
Pass Marks: 50

**ASSIGNMENT No. 1**  
(Units: 1–4)

*Note: All questions are compulsory. Each question carries equal marks.*

- Q. 1 (a) What is the object-oriented approach? What are the main characteristics of object-oriented languages?  
(b) Write down the important feature of C++ language?
- Q. 2 (a) Use for loops to construct a program that displays a diamond shape of asterisks (+) on the screen. The diamond should look like shown here, except that it should be 9 lines high, instead of 5 lines. One way to do this is to nest two inner loops, one to print spaces and one to print asterisks (+), inside an outer loop that steps down the screen from line to line.
- ```
+
+++
++++
+++
+
```
- (b) Explain the following terms: Inheritance, Polymorphism and Overloading with the help of examples.
- Q. 3 (a) Write a function that takes three distance values as arguments and returns the smallest one. Include a main ( ) program that accepts three distance values from the user, compares them, and displays the smaller.  
(b) What is the principle reason for passing arguments to a function by reference? In what unusual place can you use a function call when a function returns a value by reference?

- Q. 4 Write a program that reads a group of numbers from the user and places them in an array of type float. Once the numbers are stored in the array, the program should calculate the average of the input numbers, find the smallest and the largest number from the group of input numbers and print the result for all calculations. Use pointer notation wherever possible.
- Q. 5 Write a note on the following topics:
- Pointers
  - Reusability
  - Constructors
  - Advantages of OOP

## **ASSIGNMENT No. 2**

(Units: 5–8)

**Total Marks: 100**

**Pass Marks: 50**

*Note: All questions are compulsory. Each question carries equal marks.*

- Q. 1 (a) Compare the three access specifiers; Public, Private, and Protected. What are the major advantages and disadvantages of these access specifiers?
- (b) Explain the following term multiple inheritance with the help of at least two different examples.
- Q. 2 (a) What is meant by Operator Overloading? Explain it in detail with the help of suitable examples.
- (b) What are the major differences between Friend Functions & Static Functions? Explain them in detail.
- Q. 3 In a loop, prompt the user to enter name data consisting of a first name, middle initial, last name, and employee number (type unsigned long). Then, using formatted I/O with the insertion (<<) operator, write these four data items to an ofstream object. Don't forget that strings must be terminated with a space or other whitespace character. When the user indicates that no more name data will be entered, close the ofstream object, open an ifstream object, read and display all the data in the file, and terminate the program.
- Q. 4 (a) What is meant by throwing an exception? What is the sequence of events when an exception occurs?
- (b) Write a template function that returns the average of all the elements of an array. The arguments to the function should be the array name and the size of the array (type int). In main ( ), exercise the function with arrays of type int, long, double, and char.

Q. 5 Write a note on the following topics:

- Error Handling
  - Virtual Destructors
  - Command Line Arguments
  - Simple and Multiple Exceptions
- 

## **3407 Programming Language-I**

**Credit Hours: 4 (3+1)**

*Recommended Book:*

*The Wait Group's Object Oriented programming in C ++ 3<sup>rd</sup> Edition by Robert Lafore*

**Course Outlines:**

### **Unit No. 1 Introduction**

History of C++, Comparison of C and C++, C++ Compilers, Program Structure, Basic Input and Output Statements, Programming Exercise

### **Unit No. 2 Object Oriented Programming Concepts**

Object Oriented Approach, Objects and Classes, Characteristics of OO Languages (Inheritance, Polymorphism, Reusability, Overloading), Advantages of OOP, Programming Exercise

### **Unit No. 3 Classes and Objects**

Class definition, Class Objects, Constructors, Default Copy Constructor, Objects as Function Arguments, Functions returning Objects, Programming Exercise

### **Unit No. 4 Arrays, Pointers and Functions**

Static Memory Allocation using Arrays, Dynamic Memory Allocation and De-Allocation using Pointers, Functions returning Pointers, Function call using Pointers, Passing Pointer as a function parameter, Function overloading, Inline Functions, Programming Exercise

### **Unit No. 5 Inheritance**

Derived and Base Classes, Derived Class Constructors, Overriding Member Functions, Class Hierarchies, Public, Protected & Private Inheritance, Levels of Inheritance, Multiple Inheritance, Programming Exercise

### **Unit No. 6 Files and Streams**

Streams, String I/O, Character I/O, Object I/O, I/O With Multiple Objects, File Pointers, Disk I/O With Member Functions, Error Handling, Redirection of Input and Output, Command Line Arguments, Printer Output, Programming Exercise

### **Unit No.7 Operator Overloading &virtual Functions**

Operator Overloading (Unary Operators, Binary Operators, Data Conversion), Virtual Functions, Late binding, Abstract Classes and Pure Virtual Functions,

Virtual Destructors, Virtual Base Classes, Friend Function, Static Functions, Programming Exercise

**Unit No. 8 Templates and Exceptions Handling**

Function Templates, Class templates, Exceptions syntax, Simple and Multiple Exceptions, Exceptions with arguments, Programming Exercise

**Unit No. 9 Visual C++ Overview**

Introducing Windows Programming, Console Applications, Integrated Development Environment (The Editor, Compiler, The Linder, The Libraries, App Wizard, Class Wizard, Wizard Bar), Documentation, Projects and Workspaces, Defining a Project, Debug and Release versions, Introduction to MRF, Structure of an MFC program, Creating and Executing a Windows Program, Programming Exercise

=====