University of Central Punjab

**Faculty of Information Technology**

# Object Oriented Programming

|  |  |  |
| --- | --- | --- |
| **Lab 16** | |  |
| **Topic** | Template Function and Template Classes |
| **Objective** | The basic purpose of this lab is to implement the concept of multiple and multilevel inheritance |
|  | | |

**Instructions:**

* Indent your code.
* Comment your code.
* Use meaningful variable names.
* Plan your code carefully on a piece of paper before you implement it.
* Name of the program should be same as the task name. i.e. the first program should be Task\_1.cpp

**Students are required to work in multiple files i.e .h and .cpp**

**Task1:**

Create a function called swap () that interchanges the value of the two arguments sent to it. Make the function into a template, so it can be used with all numerical data types (int, float, char and so on) .Write main program to exercise the function with several types.

**Task 2:**

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.

**Task 3:**

Write a simple function template for predicate function isEqualTo that compares its two arguments of the same type with the equality operator (==) and returns true if they are equal and false otherwise. Use this function template in a program that calls isEqualTo only with a variety of fundamental types. Now write a separate version of the program that calls isEqualTo with a user-defined class type, but does not overload the equality operator. What happens when you attempt to run this program? Now overload the equality operator (with the operator function) operator==.Now what happens when you attempt to run this program?

**Task 4:**

Create a Class Template for class Stack.

**Functions:**

First two functions are template function:

Void Push( parameter ); // parameter should be int , float, string

ReturnType POP(); // ReturnType should be int , float, string

Void display();

Main Class should be like this:

Stack<string> obj(3);

Obj.push();

Obj.push();

Obj.push();

Obj.push();

Obj.display();

cout<<Obj.pop();

Obj.display();

Repeat the above procedure for creating a stack for integers and float.

**Task 5:**

**Class Templates:**

Create a class named as arithmetic Operations

Create functions that could ADD, SUBTRACT, MULTIPLY, DIVIDE inputs of different data types given by user.