# Government College University, Lahore

## Department of Computer Science

### Programming Fundamentals (Spring 2018)

# Lab – 06 & 07

### Week (29-04-2019 – 03-05-2019)

## Submission Deadline 12-05-2019 (Sunday) 09:00 PM

The purpose of this lab is to give students hands on experience of using Arrays. Students will also understand how to write their own functions and call them. The writing of user defined functions will allow user to make their programs more readable. This lab will give students the idea of reusability.

### Startup (01) (Understand Array Traversal)

**Type the following program in your editor and observe the results**

```
#include <iostream>

using namespace std;

main(){

     int Arr[]={12,34,54,67,89,24};

     for(x=0;x<6;x++{

          cout<<Arr[x];
     }

}
```

### Startup (02) (Understand Functions)

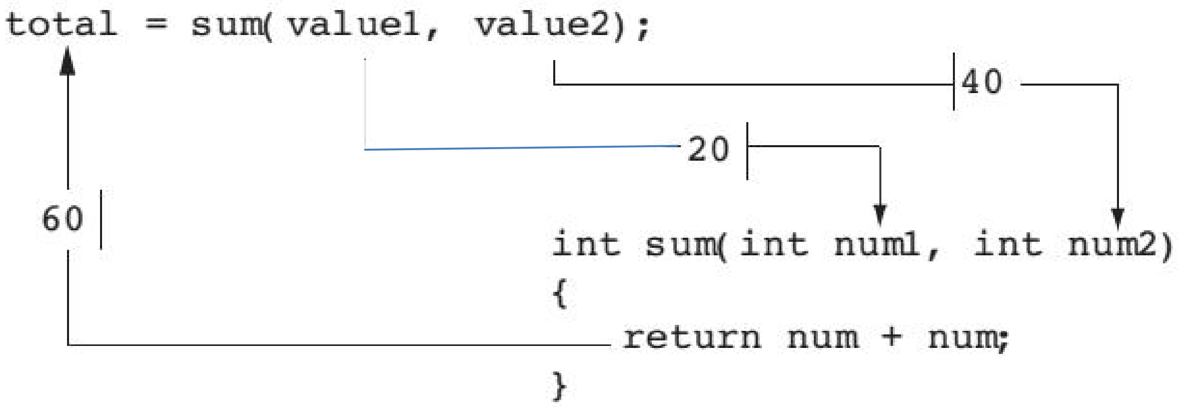
**Type the following program in your editor and observe the results**

```
#include <iostream>

using namespace std;

void displayMessage(){
     cout<<"Hello from the diplayMessage() Function.\n";
}
main(){

     cout<<"Hello from the Main.\n";

     diplayMessage();

     cout<<"Good Bye from main.";
}
```

## Understanding how function call works

This will allow student to understand how a function call works to get results from functions.

```
Total = sum (value1, value2);          Function Call
```



```
total = sum( value1, value2);

int sum( int num1, int num2)
{
    return num + num;
}
```

## Algorithmic Workbench

- *name* is an integer array with 20 elements. Write a for loop that prints each element of the array.

- The arrays *numberArray1* and *numberArray2* have 100 elements. Write a loop that copies the values from numberArray1 to numberArray2.

## Task-01 (Largest/Smallest Array Value)

Write a program that lets the user to enter 10 values into an array. The program should then display the largest and smallest value of the array.

**Hint: to input into array use cin>>Arr[index];**

## Task-02 (Rainfall Statistics)

Write a program that lets the user to enter the total rainfall for each of 12 months into an array of doubles. The program should calculate and display the total rainfall for the year, the average monthly rainfall, and the month with the highest and lowest amounts.

*Input validation: Do not accept negative numbers for monthly rainfall figures*

## Task-03 (Element Swaping)

Write a Program that will input an array of 10 integers. Your program will swap the every element with its next elements and then print it.

**Task-04 (Reverse Array Elements)**

Write a program that will reverse the elements of 1-D array. Your program will do the following.

- Ask user to size of array to insert integer elements
- Inputs array elements
- Reverse them and then print.

**Task-05 (Element Rotation)**

Write a program that will input an integer array. Your program will print the array after rotating the elements of array.

**Task-06 (Element Exchange)**

Write a program that will input an array of integer of odd size. Your program will exchange the middle element of array with the first, last element with middle and first element with last element of array and then print.

**Task-07 (Printing Even Elements)**

Write a program that inputs an array of size given by user. The program will then print only even values of arrays.

**Task-08**

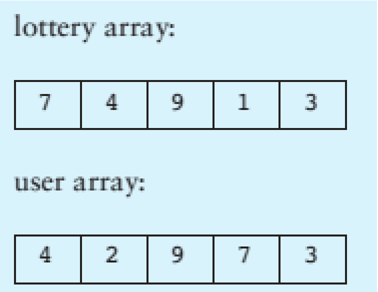
Write a program that inputs an integer array and swap first and last element of an integer 1-d array.

**Task-09 (Merging Arrays)**

Suppose A, B, C are arrays of integers of size M, N, and M + N respectively. The numbers in array A appear in ascending order while the numbers in array B appear in descending order. Write a program in C++ to produce third array C by merging arrays A and B in ascending order. Make use of nested loop where required.

**Task-10 (Lottery Application)**

Write a program that simulates a lottery. The program should have an array of five inter named lottery, and should generate a random number in the range of 0 through 9 for each element in the array. The user should enter five digits which should be stored in an integer array named *user*. The program is to compare the corresponding elements in the two arrays and keep a count of the digits that match. For example, the following shows the lottery array and the user array with sample numbers stored in each.



lottery array:

| | | | | |
|---|---|---|---|---|
| 7 | 4 | 9 | 1 | 3 |

user array:

| | | | | |
|---|---|---|---|---|
| 4 | 2 | 9 | 7 | 3 |

The program should display the random numbers stored in the lottery array and the number of matching digits. If all the digits match display a message proclaiming the user as a grand prize winner.

**Task-11 (Markup)**

Write a function that ask user to enter an item's wholesale cost and its markup percentage. It should then display the items retail price. For example

- If an items' wholesale cost is 5.00 and its markup percentage is 100%, then the items' retail price is 10.00.
- If an items' wholesale cost is 5.00 and its markup percentage is 50%, then the items' retail price is 7.50.

The program should have a function named calculateRetail that receives the wholesale cost and markup percentage as argument and return the retail price of the item.

Input validation do not accept negative values for either wholesale cost of item or percentage.

**Task-12 (Falling Distance)**

When an object is falling because of gravity, the following formula can be used to determine the distance of object falls in specific time period.

$$d = {}^{1}/2\, gt^2$$

The variables in the formula are as follows: $d$ is the distance in meters, $g$ is 9.8, and $t$ is the amount of time, in seconds, that object has been falling.

Write a function named *fallingDistance* that accepts an object's falling time (in seconds) as an argument. The function should return distance, in meters, that the object has fallen during that time interval. Write a program that demonstrates the function by calling it in a loop that passes the values 1 through 10 as arguments, and display the return value.

### Task-13 (Kinetic Energy)

In Physics, an object that is in motion is said to have kinetic energy. The following formula can be used to determine a moving objects kinetic energy:

$$KE = {}^{1}\!/_{2}\, mv^2$$

The variables in the formula are as follows: $KE$ is the kinetic energy, $m$ is the object's mass in kilogram, and $v$ is the object velocity, in meters per second.

Write a function named *kineticEnergy* that accepts an object's mass (in kilogram) and velocity (in meter per seconds) as arguments. The function should return the amount of kinetic energy that the object has. Demonstrate the function by calling it in a program that ask the user to enter values for mass and velocity.

### Task-14 (Present Value)

Suppose you want to deposit a certain amount of money into a saving account, and then leave it alone to draw interest for next 10 years. At the end of 10 years you would like to have Rs. 1,000,000 in the account. How much do you need to deposit today to make that happen? You can use the following formula, which is known as the present value formula, to find out:

$$P = \frac{F}{(1+r)^n}$$

The term in the formula are as follows:

- $P$ is the present value or the amount that you need to deposit today
- $F$ is the future value that you want in the account. ( in this case , F is Rs. 1,000,000)
- $r$ is the annual interest rate
- $n$ is the number of years that you plan to let the money sit in the account

Write a program that has a function named *presentValue* that performs this calculations. The function should accept the future value, annual interest rate, and the number of years as arguments. It should return the present value, which is the amount that you need to deposit today. Demonstrate the function in a program that lets the user experiment with different values for the formula's term.

## Task-15 (Star Search)

A particular talent competition has five judges, each of whom awards a score between 0 and 10 to each performer, Fractional score, such as 8.3, are allowed. A performers final score is determined by dropping the highest and lowest score received, then averaging the three remaining scores. Write a program that uses this method to calculate a contestant's score. It should include the following functions:

- `void getJudgeData()` should ask the user for a judge's score, store it in a reference parameter variable and validate it. This function should be called by main one for each five judges.
- `void calcScore()` should calculate and display the average of the three scores that remain after dropping highest and lowest score the performer receivesed. This function should be called just once by main, and should be passed the five scores.

The last two functions, described below, should be called by `calcScore`, which uses the returned information to determine which of the scores to drop.

- `int findLowest()` should find the lowest of the five score passed to it.
- `int findHighest()` should find the highest of the five score passed to it

**Input validation: do not accept just score below 0 and above 10.**

## Submission Instructions

- Store File as Lab-06-Task-01.cpp (Task Number will change with each task)
- Write Observation File for your lab task and store as readme.docx
- All .cpp files and .docx file must be compresses as YOUR_ROLL_NUMBER-Lab-06.zip
- Only submit one zip file

****************** BEST OF LUCK ***********