Programming Fundamentals

Lab Manual 5

Instructor:

Mr. Samyan Qayyum Wahla

Ms. Rabeeya Saleem (GA)

Learning Objectives:

- Student should be able to understand and manipulate one dimensional arrays
- Learn to write well documented and formatted code.

CLO:

- CLO1

Registration Number:

_____

Name:

_____

Guidelines/Instructions:

- Use of VS Code is must in this lab.
- Write well commented code.
- Name of variables should be meaningful.
- Code should be well formatted.
- Create meaningful variable names. Add comments for readability. Indent each line of your code.
- Plagiarism/Cheating is highly discouraged by penalizing to both who tried and one who shared his/her code.

Reading Content:

Chapter 8 of C++: Programming from Problem Analysis to Program Design by D.S Malik

- From start to article Some Restrictions of Array Processing
- From C-String to String Output

Arrays:

Array is the collection of variables of same data type also known as Homogeneous data type. Each array has the following attributes

- Name of the array (In the similar fashion of identifier)
- Array data type
- Size of the array

Array Declaration:

```
int intArray[10];        //declares the array of integers of size 10
double array2[5];        //declares the array of double of size 5
char array3[10];         //declares the array of characters of size 10
```

To Explore:

Does c++ allow variable length sized array

E.g. if we declare the array as following:

```
int size =10;
int intArray[size];
```

Array Manipulation for int, double and float data type:

Example 1: Input the 7 numbers from user and print them on screen.

```
int array[7];

//input into array
for (int i=0; i < 7; i++){
    cout<<"Enter a number: ";
    cin>>array[i];
}

//output of the array
for (int i=0; i < 7; i++){
    cout<<array[i]<<endl;

}
```

Example 2: Input the 5 numbers from user and copy them into another array.

```
int array1[5];
int array2[5];

//input into array1
for (int i=0; i < 5; i++)
{
    cout<<"Enter a number: ";
    cin>>array1[i];
}

//copy of the array1 into array2
for (int i=0; i < 5; i++)
{
    array2[i]=array1[i];
}

//output of the array2
for (int i=0; i < 5; i++)
{
    cout<<array2[i]<<endl;

}
```

Array Manipulation char data type:

Example 3: Input the string without space.

```
char array[20];
cout<<"Enter a string:";
cin>>array;

cout<<"Entered string is "<<array<<endl;
```

Example 4: Input the string with space.

```
   char array[20];
   cout<<"Enter a string:";
   cin.get(array,13);

   cout<<"Entered string is "<<array<<endl;
```

Example 5: copy the character array

```
#include <iostream>
#include<cstring>   //give the functions for the character array(See the details in book)

using namespace std;
int main()
{
   char array1[20];
   char array2[20];
   cout<<"Enter a string:";
   cin>>array1;


   int arraySize= strlen(array1);   //strlen return the length of string entered by users
   cout<<"Size of the array: "<<arraySize<<endl;
   for(int i = 0 ; i<arraySize ; i++)
   {
       array2[i]=array1[i];
   }

   array2[arraySize] = '\0';   //remove this line the write down the effect on output



   cout<<"Entered string is "<<array1<<endl;
   cout<<"Copied string is "<<array2<<endl;
    return 0;
}
```

Parallel Arrays:

Example 6: Input the marks of 5 students from user and display them.

```
#include <iostream>

using namespace std;
int main()
{
    string student[5]; //student and marks are parallel
    int marks[5];      //arrays, notice the same size of both

    for (int i = 0; i < 5; i++)
    {
        cout << "Enter Name of student " << i + 1 << ": ";
        getline(cin,student[i]);  //for input of spaces into string data type

        cout << "Enter Marks of student " << i + 1 << ": ";
        cin >> marks[i];

        cout<<endl;
        cin.ignore(25,'\n');
    }

    cout<<"Student Name\tMarks"<<endl;
    for(int i = 0 ; i < 5; i++)
    {
        cout<<student[i]<<"\t"<<marks[i]<<endl;
    }

    return 0;
}
```

Output

```
Enter Name of student 1: Muhammad ALi
Enter Marks of student 1: 34

Enter Name of student 2: Ali Ahmed
Enter Marks of student 2: 43

Enter Name of student 3: Samyan Wahla
Enter Marks of student 3: 75

Enter Name of student 4: Ali Imran
Enter Marks of student 4: 74

Enter Name of student 5: Ahmed Ali
Enter Marks of student 5: 12

Student Name    Marks
Muhammad ALi    34
Ali Ahmed       43
Samyan Wahla    75
Ali Imran       74
Ahmed Ali       12
```

Remove cin.ignore and write down the effect on output.

Why 25 is written in cin.ignore, can we write any other number? What are the possibilities.

Example 7: Input the marks of 5 students from user and display their weightage out of 40 along with marks.

```
#include <iostream>
#include<iomanip>
using namespace std;
int main()
{
    string student[3]; //student and marks are parallel
    int marks[3];      //arrays, notice the same size of both

    for (int i = 0; i < 3; i++)
    {
        cout << "Enter Name of student " << i + 1 << ": ";
        getline(cin,student[i]);  //for input of spaces into string data type

        cout << "Enter Marks of student " << i + 1 << ": ";
        cin >> marks[i];

        cout<<endl;
        cin.ignore(25,'\n');

    }

    cout<<"Student Name\tMarks\tWeightage"<<endl;
    for(int i = 0 ; i < 3; i++)
    {
        double weightage = marks[i]/100.0*40;
        cout<<student[i]<<"\t"<<marks[i]<<"\t"<<fixed<<setprecision(2)<<weightage<<endl;
    }

    return 0;
}
```

Output

```
Enter Name of student 1: Muhammad Ali
Enter Marks of student 1: 54

Enter Name of student 2: Samyan Wahla
Enter Marks of student 2: 73

Enter Name of student 3: Muhammad Zain
Enter Marks of student 3: 79

Student Name    Marks   Weightage
Muhammad Ali    54      21.60
Samyan Wahla    73      29.20
Muhammad Zain   79      31.60
```

- Remove fixed and see the difference on output.
- What is the effect of setprecisison?
- What is the effect of removing #include<iomanip>

Lab Tasks:

1. Write a program to take 10 integers from user and display each integer on separate line in reverse order.

Sample Input:	**Sample Output:**
Enter Number 1: 5	1
Enter Number 2: 6	10
Enter Number 3: 4	53
Enter Number 4: 3	43
Enter Number 5: 2	23
Enter Number 6: 23	2
Enter Number 7: 43	3
Enter Number 8: 53	4
Enter Number 9: 10	6
Enter Number 10: 1	5

2. Write a program to find the sum and product of all elements of an array of type integer of size 5.

Note: Take the input in one loop and then write a separate loop to calculate sum and product.

Sample Input:	**Sample Output:**
Enter Number 1: 5	Sum of all elements is: 20
Enter Number 2: 6	Product of all elements is: 720
Enter Number 3: 4	
Enter Number 4: 3	
Enter Number 5: 2	

3. Take an array of 10 elements. Split it into middle and store the elements in two different arrays. E.g.-

Initial array :

58	24	13	15	63	9	8	81	1	78

After splitting :

58	24	13	15	63
9	8	81	1	78

4. Find largest and smallest elements of an array.

Sample Input:	**Output:**
Enter size of array: 5	Smallest Element: 1
Enter array elements: 2	Largest Element: 7
Enter array elements: 3	
Enter array elements: 1	
Enter array elements: 7	
Enter array elements: 4	

5. Write a C++ program to find the element that appears once in an array of integers.

Sample Input:	**Output:**
Enter the size of array: 5	Element of array that appear once in an array: 5 7
Enter array elements 1: 4	
Enter array elements 2: 3	
Enter array elements 3: 5	
Enter array elements 4: 7	
Enter array elements 5: 4	

6. Write a C++ program to update every array element by multiplication of next and previous values of a given array of integers

Sample Input:	**Output:**
Enter the size of array: 5 Enter array elements 1: 2 Enter array elements 2: 3 Enter array elements 3: 4 Enter array elements 4: 5 Enter array elements 5: 6	Output array is 6 8 15 24 30

7. Sort the array of 10 integers taken from user.

Sample Input:	**Output:**
Enter the size of array: 10 Enter array element 1: 12 Enter array element 2: 17 Enter array element 3: 1 Enter array element 4: 4 Enter array element 5: 8 Enter array element 6: 3 Enter array element 7: 15 Enter array element 8: 16 Enter array element 9: 6 Enter array element 10: 2	Sorted array is 1 2 3 4 6 8 12 15 16 17

8. **Number Validator**: Take a string from the user and tells whether a string is number or not

Sample Input:	**Output:**
Enter the string : 127	User enter integer
Enter the string : Programming	User enter string

9. **Float Validator:** Take a string from the user and tells whether a string is double or not

Sample Input:	**Output:**
Enter the string : 127	User entered value is not double
Enter the string : Programming	User entered value is not double
Enter the string : 127.57	User entered value is double

10. Write a C++ program to separate even and odd numbers of an array of integers. Put all even numbers first, and then odd numbers.

Sample Input	**Sample Output**
`Enter Array Elements: 12 17 1 4 8 3 15 16 6 2`	`Separated array is: 2 4 6 8 12 16 1 3 15 17`

Home Tasks:

1. Write a C++ program to move all negative elements of an array of integers to the end of the array without changing the order of positive element and negative element. Size of the array is 10.

Sample Input	**Sample Output**
`Enter Array Elements: 1 -1 3 2 -7 -5 11 6`	`1 3 2 11 6 -1 -7 -5`

2. Write a program that initializes an array. It inputs a value from the user and searches the number in the array. Finally, it will display the location of number in array if it exists. (using sequential search)

Sample Input	**Sample Output**

`Enter the size of array: 6`
 	
`Enter Array Elements: 2 6 3 5 4 9`
	`5 exist in given array`
	`Location of searched number is 3 index of array`

Enter Element to Search: 5	

3. **Email Validator:** Write a program to validate a string whether it is valid email or not

Sample Input	**Sample Output**
Enter Email Address: programming2020@testcom	Invalid Email address
Enter Email Address: programming2020@test.com	Valid Email address

4. Write a program that asks the user to type 10 integers of an array. The program will then display either "the array is growing", "the array is decreasing", "the array is constant", or "the array is growing and decreasing."

Sample Input	**Sample Output**
Enter Array Elements: 1 2 3 4 5 6 7 8 9	Array is increasing
Enter Array Elements: 9 8 7 6 5 4 3 2 1	Array is decreasing
Enter Array Elements: 3 3 3 3 3 3 3 3 3	Array is constant
Enter Array Elements: 1 2 3 4 5 6 3 2 1	Array is increasing and decreasing

5. Write a program that uses three arrays **Mango**, **Orange** and **Banana** to store the number of fruits purchased by 3 customers. The program inputs the number of mangos, oranges and bananas to be purchase by customer and stores them in corresponding array. The program finally displays the total bill of each customer according to the following prices:

   Rs. 20 per mango
   Rs. 10 per orange
   Rs. 5 per banana

Sample Input

Enter Number of Customer: 3
Customer 1:
Enter Number of Mangoes : 2
Enter Number of Oranges : 3
Enter Number of Banana : 4
Customer 2:
Enter Number of Mangoes : 7
Enter Number of Oranges : 9
Enter Number of Banana : 1
Customer 3:
Enter Number of Mangoes : 3
Enter Number of Oranges : 5
Enter Number of Banana : 7

Sample Output:

****

Customer No.	**Mangoes**	**Oranges**	**Bananas**	**Total Bill**
1	2	3	4	Rs.90
2	7	9	1	Rs. 235
3	3	5	7	Rs. 145

Challenge Yourself: (Optional Part)

1. Write a C++ program to find the element that appears once in an array of integers.

Sample Input:	**Sample Output:**
Enter the size of array: 5 Enter array element 1: 4 Enter array element 2: 3 Enter array element 3: 5 Enter array element 4: 7 Enter array element 5: 4	Element of array that appear once in an array: 5 7

2. Write a program that uses four arrays **numbers, squares**, **and cubes** and **sums** each consisting of 5 elements. The **numbers** array stores the values of its indexes, the **squares** array stores the squares of its indexes, the **cubes** array stores the cubes of tits indexes and **sums** array stores the sum of corresponding indexes of three arrays. The program should display the values of all arrays and the total of all values in **sums** array.

Sample Input	**Sample Output:**
Enter Array Elements: 0 1 2 3 4	Numbers: 0 1 2 3 4 Squares: 0 1 4 9 16 Cubes: 0 1 8 27 64 Sums: 0 3 14 39 84 Grand total: 140

3. Write a program that initializes an array. It inputs a value from the user and searches the number in the array. Finally, it will display the location of number in array if it exists. (using binary search)

Sample Input	**Sample Output**

`Enter the size of array: 6`
 	
`Enter Array Elements: 2 6 3 5 4 9`
 	
`Enter Element to Search: 5`
	`5 exist in given array`
	`Location of searched number is 3 index of array`

4. **Registration Number Validator:** Write a program to validate a string whether it is valid registration number or not.

Sample Input	**Sample Output**
`Enter Registration Number: 2020cS111`	`Invalid Registration Number`
`Enter Registration Number: 2018-CS-111`	`Valid Registration Number`

5. Write a program that initializes an array of integers having at-least 10 elements. Write a program to find those pair of elements that has the maximum and minimum difference among all element pairs

Sample Input	**Sample Output**

`Enter the size of array: 6`
 	
`Enter Array Elements: 2 3 10 6 4 8 1`
	`Maximum Difference: 8`
	`Minimum Difference: 1`

What to Submit:

1. Lab Tasks should be named from Task1.cpp to Task10.cpp and submit in LabManual5 LabTasks Assignment on Eduko in a zip folder.
2. Home Tasks should be named from Task1.cpp to Task5.cpp and submit in LabManual5 HomeTasks Assignment on Eduko in a zip folder.
3. Challenge Yourself is optional part. You can submit those from Task1.cpp to Task5.cpp in LabManual5 Challenge Assignment on Eduko in a zip folder.