Programming Fundamental

Lab Manual - Week 09

Introduction

Welcome Back to your favorite Programming Lab students. In this lab manual, we shall work together to learn and implement new programming concepts.

Skills to be learned:

- Declare and initialize arrays of different data types.

Let's do some coding.

Skill: Declare and initialize arrays of different data types.

Introduction

By this week, you have learned how to write a program that contains functions, loops, and conditional structures. In this class, we will learn about another very powerful concept known as Arrays.

An array is a collection of similar data items stored at contiguous memory locations and elements that can be accessed randomly using the indices of an array. Following are the types of arrays that are often used in programs.

- Integers arrays
- Float arrays
- Char arrays
- String arrays

Consider a task in which you want to store the mid-term marks of Programming Fundamentals of 200 students. Let's create an Array to store the marks of 200 students of the Computer Science Department of UET Lahore.

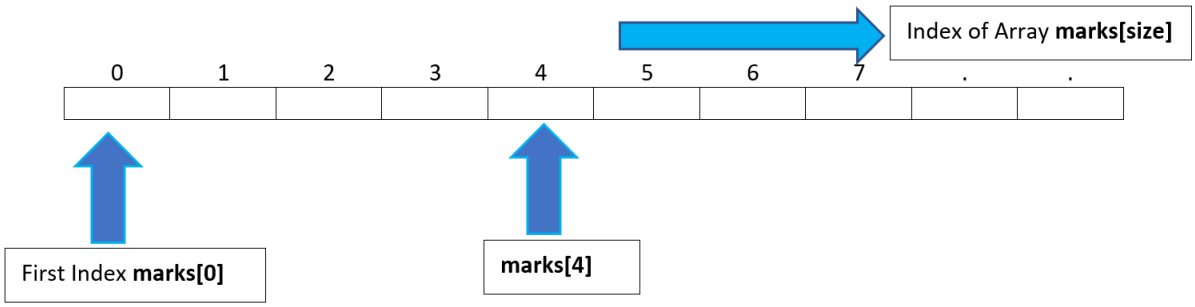
Array Declaration

Following is the Syntax of an Array **dataType ArrayName[size];**	For example, int marks[200];
This c++ statement will declare an array of **size** in the memory of the computer. Now,	

we can use the indices of the array to store the marks of the students at different addresses.

Index of Array

An array is a collection of items on contiguous memory locations. Every item in the array exists on a unique address in memory. Consider the following diagram for a better understanding.



We can use the assignment operator to assign a value using the index of the array item.

Consider the following program for better understanding

Task 01(WP): Write a program that declares an array of 5 integer elements, initializes them one by one and displays them.

```
main()
{
    int numbers[5];

    numbers[0] = 15;
    numbers[1] = 12;
    numbers[2] = 13;
    numbers[3] = 19;
    numbers[4] = 10;

    cout << "The 1st element at location numbers[0] is: " << numbers[0] << endl;
    cout << "The 2nd element at location numbers[0] is: " << numbers[1] << endl;
    cout << "The 3rd element at location numbers[0] is: " << numbers[2] << endl;
    cout << "The 4th element at location numbers[0] is: " << numbers[3] << endl;
    cout << "The 5th element at location numbers[0] is: " << numbers[4] << endl;

}
```

We use the assignment operator to assign values to array item using indices of an array.

Skill: Declare and initialize arrays of different data types

```
The 1st element at location numbers[0] is: 15
The 2nd element at location numbers[0] is: 12
The 3rd element at location numbers[0] is: 13
The 4th element at location numbers[0] is: 19
The 5th element at location numbers[0] is: 10
```

Task 02(CA): Write a program that declares an array of 5 elements, initializes them one by one and displays the 2nd and 4th elements of an array

Similarly, we can store the value entered by the user in the array by using the following statement.
cin >> arrayName[index];

Task 03(WP): Write a program that declares an array of 5 elements, initializes them one by one by user input, and displays 1st and last elements of the array.

```
main()
{
    int numbers[5];
    cout << "Enter 1st Number: ";
    cin >> numbers[0];
    cout << "Enter 2nd Number: ";
    cin >> numbers[1];
    cout << "Enter 3rd Number: ";
    cin >> numbers[2];
    cout << "Enter 4th Number: ";
    cin >> numbers[3];
    cout << "Enter 5th Number: ";
    cin >> numbers[4];

    cout << "The 1st element in array is: " << numbers[0] << endl;
    cout << "The last element in array is: " << numbers[4] << endl;

}
```

Skill: Declare and initialize arrays of different data types

```
Enter 1st Number: 12
Enter 2nd Number: 14
Enter 3rd Number: 15
Enter 4th Number: 13
Enter 5th Number: 17
The 1st element in array is: 12
The last element in array is: 17
```

🚧 Problem 🚧

We have used the same variable name to store multiple values but our code is repeating for taking input.
Is there any better way ❓

💡 Solution 💡

We can use the counting loop for taking input in the array from the user?

Consider the same question with loops below.

Task 04(WP): Write a program that declares an array of 5 elements, initialize them one by one by user input and display 1st and last elements of the array.
Instruction: Use a loop for taking input from the user

```
main()
{
    int numbers[5];

    for (int count = 0; count < 5; count = count + 1)
    {
        cout << "Enter Number: ";
        cin >> numbers[count];
    }
    cout << "The 1st element in array is: " << numbers[0] << endl;
    cout << "The last element in array is: " << numbers[4] << endl;
}
```

```
Enter 1st Number: 12
Enter 2nd Number: 14
Enter 3rd Number: 15
Enter 4th Number: 13
Enter 5th Number: 17
The 1st element in array is: 12
The last element in array is: 17
```

Skill: Declare and initialize arrays of different data types

Programming Fundamental

Lab Manual - Week 09

Task 05(OP): Write a program that takes **n numbers** from the user, stores them in an array, and displays those numbers on the screen.

Similarly, we can use the array for solving more complex problems.
Consider the following task for better understanding.

Task 06(CL): Write a program that prints the sum and average of the first 5 natural numbers on the screen.

```
main()
{
    int sum = 0;
    float average = 0;
    int numbers[5] = {1, 2, 3, 4, 5};
    for (int idx = 0; idx < 5; idx = idx + 1)
    {
        sum = sum + numbers[idx];
    }
    average = sum/5;
    cout << "Sum: " << sum << endl;
    cout << "Avg: " << average << endl;
}
```

```
Sum: 15
Avg: 3
```

String as Character Array

String values are stored as a character array inside memory and the end of the string is stored as '/0' referred to as a null character.

Consider the following task for better understanding.

Task 07(WP): Write a program that prints whether a specific character is present in a string or not.

Skill: Declare and initialize arrays of different data types

```
main()
{
    string word;
    char letter;
    bool isFound = false;

    cout << "Enter a Word: ";
    cin >> word;
    cout << "Enter the character you want to find: ";
    cin >> letter;

    for (int idx = 0; word[idx] != '\0'; idx = idx + 1)
    {
        if (word[idx] == letter)
        {
            isFound = true;
            break;
        }
    }
    if (isFound == true)
    {
        cout << letter << " is found in " << word;
    }
    else
    {
        cout << letter << " is not found in " << word;
    }
}
```

```
Enter a Word: programming
Enter the character you want to find: m
m is found in programming
```

Consider the following amazing facts about the string variables.

Do you know? 💡
You can use the **getline(cin, stringVariableName)** function to take string input that includes a space character.

Do you know? 💡
You can declare a string by using any of the following methods. Pretty cool, isn't ?

Skill: Declare and initialize arrays of different data types

```
main()
{
    string word = "C++";
    char word[4] = "C++";
    char word[] = "C++";
    char word[]= {'C','+','+','\0'};
    char word[4]= {'C','+','+','\0'};
    char word[]= {'C','+','+'};
    char word[100] = "C++";
}
```

Task 08(CA): Write a program that stores a word in an array and displays the location of all alphabets in the array.

Task 09(OP): Write a C++ program that is given a string as input, it displays true if its length is even and false if the length is odd.

Task 10(CA): Write a program that takes a string from the user and displays it in reverse order.

Task 11(OP): Create a C++ program that takes a string (a random name) and a letter from the user. If the last character of the name is is same as the letter, it displays true, otherwise returns false.

Task 12(CP): Write a C++ program to change every letter in a given string with the letter following it in the alphabet (ie. a becomes b, p becomes q, z becomes a).
For Example:
Input: aslam
Output: btmbn

Task 14(CP):
Write a c++ program to count all the vowels in a given string.
Example:
Sample Input: eagerer
Sample output: number of vowels: 4

Skill: Declare and initialize arrays of different data types

Task 15(OP):

Write a program that takes **n** numbers from the user, stores them in an array, and prints them in reverse order.

Task 16(OP):

Write a program that takes **n** numbers from the user and stores them in an array.
The program should also ask for another number from the user and print "Already Entered" if the user has already entered that number.

Task 17(OP):

Write a program that takes **n** numbers from the user and stores them in an array.
The program should also ask for another number and print the scalar product of that number with each of the n numbers.

Task 18(OP):

Write a program that takes **n** numbers from the user and stores them in an array and prints the largest number entered by the user.

Task 19(OP):

Write a program that takes **n** numbers from the user and stores them in an array and prints the smallest number entered by the user.

Task 20(CP):

When resistors are connected together in series, the same current passes through each resistor in the chain and the total resistance, RT, of the circuit must be equal to the sum of all the individual resistors added together. That is
RT = R1 + R2 + R3 ...
Create a program that takes an array of values resistance that are connected in series, and calculates the total resistance of the circuit in ohms. The ohm is the standard unit of electrical resistance in the International System of Units (SI).

Test Cases:

[1, 5, 6, 3] ➞ "15 ohms"
[16, 3.5, 6] ➞ "25.5 ohms"
[0.5, 0.5] ➞ "1.0 ohms"

Skill: Declare and initialize arrays of different data types

Task 21(CP):

Create a program that takes two arrays and inserts the second array in the middle of the first array. The first array always has two elements.

Test Cases:

[1, 10], [2, 3, 4, 5, 6, 7, 8, 9] ➞ [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
[15,150], [45, 75, 35] ➞ [15, 45, 75, 35, 150]

Task 22(CP):

Given a total due and an array representing the amount of change in your pocket, determine whether or not you are able to pay for the item. The change will always be represented in the following order: quarters, dimes, nickels, and pennies.

[25, 20, 5, 0], 4.25 should yield true, since having 25 quarters, 20 dimes, 5 nickels, and 0 pennies give you 6.25 + 2 + .25 + 0 = 8.50.

NOTE:

- quarter: 25 cents / $0.25
- dime: 10 cents / $0.10
- nickel: 5 cents / $0.05
- penny: 1 cent / $0.01

Test Cases:

[2, 100, 0, 0], 14.11 ➞ false
[0, 0, 20, 5], 0.75 ➞ true
[30, 40, 20, 5], 12.55 ➞ true
[10, 0, 0, 50], 3.85 ➞ false
[1, 0, 5, 219], 19.99 ➞ false

Task 23(CP):

Write a program that prints the string "something" joined with a space " " and the given argument a.

Test Cases:

"is better than nothing" ➞ "something is better than nothing"
"Bob Jane" ➞ "something Bob Jane"
"something" ➞ "something something"

Task 24(CP):

Create a program that takes a string and returns a new string with all vowels removed.

Skill: Declare and initialize arrays of different data types

Test Cases:

Input:

"I have never seen a thin person drinking Diet Coke."

Output:

" hv nvr sn thn prsn drnkng Dt Ck."

Good Luck and Best Wishes !!

Happy Coding ahead :)