DHA Suffa University
CS 1001L – Programming Fundamentals Lab
Fall 2020

Lab 09 – Arrays

Objective(s):

- To learn how to use Arrays.
- To learn using For loop for Array access.
- To learn which problems to use arrays on.

Array

An array is a **series of elements** of the **same type** placed in **contiguous memory locations** that can be **individually referenced** by adding an index to a **unique identifier**.

Declaring Arrays

In order to declare 4 variables, more specifically - integers, using an array we write:

int nums[4];

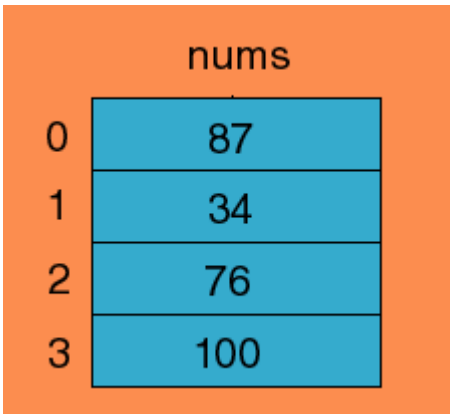
- This sets up in memory 4 integers, i.e. 4 consecutive boxes in memory, each box holding an integer.
- We say that *nums* is an array of type integer or simply, an integer array.
- Each integer in the array is called an element of the array. In this example, each individual element is simply an integer.

Accessing Array Elements:

Each array element is located at a unique index and hence, we can access a particular element using the corresponding index number. For an array of size n, the **indices always go from 0 to n-1**. Consider the same e.g. of *nums* which is an integer array of 4 elements. Since there are 4 elements (i.e. n=4) so the indices will be **0, 1, 2 and 3** and at each index there will be an integer value:

```
nums[0] = 87;
nums[1] = 34;
nums[2] = 76;
nums[3] = 100;
```

Here, the number 87 is the first element of the array and is located at index 0. Similarly, 34 is the second element and is at index 1, and so on. So this is what'll be happening in memory:



Example 01: Display the values of **Array** marks without using loop.
Pseudocode:

```
1.  print array[0]
2.  print array[1]
3.  print array[2]
4.  print array[3]
5.  print array[4]
```

Program in C:

```c
#include<stdio.h>

int main()
{
    int marks[5] = {89, 65, 87, 80, 70};
    printf("Marks for Subject 1: %d \n", marks[0]);
    printf("Marks for Subject 2: %d \n", marks[1]);
    printf("Marks for Subject 3: %d \n", marks[2]);
    printf("Marks for Subject 4: %d \n", marks[3]);
    printf("Marks for Subject 5: %d \n", marks[4]);

    return 0;
}
```

Task01 (task1.c): Write a program to display the everyday temperature of whole week.

Loops and 1 Array

Just like any other variable that is declared and not initialized, array elements also hold garbage values if not initialized. Initializing arrays can be done generally in two ways:

int nums[4] = {0,0,0,0}; //as seen in example 01

OR

using a loop // (for loop should be our choice, why?)

Example 02: Display the square of array elements using for loop.
Pseudocode:

```
1.  for i=0 to 4
        input array[i]
2.  for i=0 to 4
        square[i] = array[i]*array[i]
        print square[i]
```

Program in C:

```c
#include<stdio.h>

int main()
{
    int array[5];
    int square[5];
    int i = 0;

    for(i=0; i<5; i++)
    {
        printf("Enter the Value for Array:");
        scanf("%d",&array[i]);
    }
    for(i=0; i<5; i++)
    {
        square[i] = array[i] * array[i];
        printf("Square of Array Element %d is \t %d \n", array[i], square[i]);
    }
    return 0;
}
```

Task02 (task02.c): Write a C program to Calculating an Average of Array Elements

Example 03: Identify the elements of an array are even or odd using for loop;
Pseudocode:

```
1.  for i=0 to 9
        input array[i]
2.  for i=0 to 9
    1.  if(array[i]%2==0)
            print "Even Number in an Array"
    2.  else
            print "Odd Number in an Array"
```

Program in C:

```c
#include<stdio.h>

int main()
{
    int array[10];
    int i = 0;
    for(i=0; i<10; i++)
    {
        printf("Enter the Value for Array:");
        scanf("%d",&array[i]);
    }
    for(i=0; i<10; i++)
    {
        if(array[i]%2==0)
        {
            printf("%d is Even Number in an Array \n", array[i]);
        }
        else
        {
            printf("%d is Odd Number in an Array \n", array[i]);
        }
    }
    return 0;
}
```

Task03 (task3.c): Write a C program to find the maximum of 10 array elements.
E.g. Input: {1, 6, 5, 9, 6, 65, 84, 0, 62, 6} Output: 84
Input: {0, 0, 5, 0, 6, -5, 4, 0, 12, 8} Output: 12

Example 04: Find the average of marks of the student in the first semester result using arrays with functions

Pseudocode:

```
1.  for i=0 to 4
        Input marks[i]
2.  average = getPercentage(marks, 5)
3.  Output average


function getPercentage(array[], arraySize)
1.  for i=0 to arraySize
        sum = sum + array[i]
2.  average = sum / arraySize
3.  return average
```

Program in C:

```c
#include<stdio.h>
float getAverage(int array[], int arraySize);
int main()
{
    int marks[5] = {0,0,0,0,0};
    int i=0;
    float average = 0.0;
    for(i=0; i<5; i++)
    {
        scanf("%d",&marks[i]);
    }
    average = getAverage(marks, 5);
    printf("An Average marks of the student in the first semester is: %.2f", average);
    return 0;
}

float getAverage(int array[], int arraySize)
{
    int i = 0;
    int sum = 0;
    float average = 0.0;
    for(i=0; i<arraySize; i++)
    {
        sum = sum + array[i];
    }
    average = sum / arraySize;
    return average;
    }
```

Task04 (task4.c): Write a C program to calculate the average temperature of the current week using array with functions.

Lab Assignment 09

Problem 01 (ascendingOrder.c): Write a C program to convert an array into ascending order.

Test Case 01:

Sample Inputs:

```
Enter the 1 Element of an Array1
Enter the 2 Element of an Array6
Enter the 3 Element of an Array5
Enter the 4 Element of an Array4
Enter the 5 Element of an Array2
```

Sample Outputs:

```
Element of an original Array are:1 ,6 ,5 ,4 ,2 ,
Element of an Array after sorting are:1 ,2 ,4 ,5 ,6 ,
```

Problem 02 (searchInArray.c): Write a C program to search an element entered by user from array and display the searched element and its location.

Test Case 01:

Sample Inputs:

```
Enter the 1 Element of an Array: 2
Enter the 2 Element of an Array: 4
Enter the 3 Element of an Array: 6
Enter the 4 Element of an Array: 4
Enter the 5 Element of an Array: 9
Element of an Array are:2 ,4 ,6 ,4 ,9 ,
Enter the Element to Search in an Array:6
```

Sample Output:

```
Element 6 found at 2 Location
```

Test Case 02:

Sample Inputs:

```
Enter the 1 Element of an Array: 9
Enter the 2 Element of an Array: 6
Enter the 3 Element of an Array: 8
Enter the 4 Element of an Array: 1
Enter the 5 Element of an Array: 7
Element of an Array are:9 ,6 ,8 ,1 ,7 ,
Enter the Element to Search in an Array:11
```

Sample Output:

```
Element Not Found
```

Lab Assignment Submission Instructions:

1. **Number your c files as question number e.g. Q1.c, Q2.c, etc. (Q is in upper case)**
2. **Create a new folder named cs2010abc where abc is your 3 digit roll #. e.g. cs201001.**
3. **Copy all the files into this folder.**
4. **Right-click on the folder you created and create a zip file by selecting the option**
   - **"Send to" and selecting "Compressed (zipped) folder" [for windows].**
   - **"Create Archive" and change option to ".zip" instead of ".tar.gz" and click on "Create". [for linux]**

   Now make sure a zip file named cs201abc.zip is created e.g. cs201001.*zip*.
5. **Submit on LMS**