

#11 Arrays

By Saurabh Shukla | 2017©mysirg.com

What is an Array?

- Arrays are the collection of a finite number of homogeneous data elements.
- Elements of the array are referenced respectively by an index set consisting of n consecutive numbers and are stored respectively in successive memory locations.
- The number n of elements is called the length or size of the array.

Think about a scenario where you need to handle hundreds of variables or even more than that. In such scenario, you might be thinking about what variable names should be used, how to reduce redundant code, etc

Let's say, you have to store marks of 100 students. Think about the following:

- 1) What should be your variable naming convention?
- 2) How you can efficiently write input instruction to store 100 data.
- 3) How could you easily manipulate data like adding all of them in a less complex style?

The answer to all these questions is subscript notation also known as Arrays.

- Array is a collection of similar elements. These similar elements could be all ints, floats, doubles, chars, etc. Array is also known as subscript variable.
- Array can store a fixed-size sequential collection of elements of the same type.
- Array elements are stored in contiguous memory block.
- Instead of declaring individual variables, such as number0, number1, ..., and number99, you can declare one array variable.
- Creating array is creating a group of variables and all its elements are accessed via single name and different subscript values.

Properties of Array:

- 1) Indexing of an array begins from zero (0).
- 2) The variable name of array contains the base address of the memory block.
- 3) The array variable are created at the time of compilation.
- 4) The size of the array cannot be altered at runtime.

Array Declaration

To declare an array in C, a programmer specifies the type of the elements and the number of elements required by an array as follows

```
<data type> arrayName [arraySize];
```

This is called single dimension Array.

The arraySize must be an integer constant greater than zero and type can be any valid C data type.

For example:

```
int a[5];
```

```
float k[4];
```

Initializing array

- `int a[5]={34,56,78,99,12};`
- `int a[5]={34,56};`

Initializing an array with lesser values than its size is allowed and the remaining array blocks will contains zero.

- `int a[5]={34,56,78,99,12,44,66};`

We cannot initialize an array with values exceeding its size.

- `int a[]={34,56,78,99,12};`

We can declare an array without mentioning its size only when initializing array during declaration.

- `int a[5];`

In the above declaration 5 is the size of an array or we can say an array is a collection of 5 variables.

Variables in an array do not have names. These variables can be recognized by their position number in an array known as index. Index of the very first variable is 0 (zero). So the index of last variable is 4 (according to our example). To assign value in any particular variable of an array we have to follow the syntax:

```
a[index]=value;
```

For example:

```
a[0]=34;
```

```
a[1]=21;
```

It is worth mentioning here that the value written in square bracket is size of array during declaration and it is index afterwards.

Example: Program to calculate average of 10 numbers (say marks)

```
1  #include<stdio.h>
2  #include<conio.h>
3  int main()
4  {
5      int i, marks[10], sum=0;
6      float avg;
7      printf("Enter 10 numbers");
8      for(i=0;i<=9;i++)
9          scanf("%d",&marks[i]);
10
11     for(i=0;i<=9;i++)
12         sum=sum+marks[i];
13     avg=sum/10.0;
14     printf("Average is %f",avg);
15     getch();
16     return(0);
17 }
```

- Total numbers of variables in this program are 13. Their names are i, sum, avg, marks[0], marks[1],...marks[9].
- The input statement, scanf() is repeated 10 times, this is possible as we can access array index with the help of variable. Observe &marks[i] in scanf(), here i is used for indexing. As the loop proceeds, value of i changes from 0 to 9 (one by one). In this way we need not to write scanf() 10 times.
- Data manipulation becomes also easy, we added all 10 values stored in an array putting statement sum=sum+marks[i] in the loop.
- Since we have an array of size 10 and each of these blocks are of type int, total memory consumed for this array is 40 bytes (4 bytes for each).

About memory Allocation of an array

Whatever the size of an array, it always consumes memory in contiguous manner.

- `int a[4];`

In the above declaration, array size is 4 which mean an array is a collection of 4 int type variables.

Thus the total memory occupied is 16 bytes

Important Note

- The array elements can be accessed in a constant time by using the index of the particular element.
- To access an array element, address of an element is computed as an offset from the base address of the array and one multiplication is needed to compute what is supposed to be added to the base address to get the memory address of the element
- First the size of an element of that data type is calculated and then it is multiplied with the index of the element to get the value to be added to the base address.

- This process takes one multiplication and one addition. Since these two operations take constant time, we can say the array access can be performed in constant time.

References:

YouTube video links

- Lecture 11 Arrays in C part-1
 - https://youtu.be/lc3RtR_345g?list=PL7ersPsTyYt2Q-SqZxTA1D-melSfqBRMW
- Lecture 11 Arrays in C part-2
 - <https://youtu.be/pWXXnUliR8I?list=PL7ersPsTyYt2Q-SqZxTA1D-melSfqBRMW>
- Why array index always begins from 0
 - <https://youtu.be/GhNtdjdSJVQ?list=PL7ersPsTyYt2sohwBWF03kSpW1Ymt0Q9R>

Exercise

- 1) Write a program to find greatest among 10 integers. Use array to store integers.
- 2) Write a program to find smallest among 20 integers. Use array to store integers.
- 3) Write a program to find an element in an array of size 10.
- 4) Write a program to find second largest element in an array of size 10.
- 5) Write a program which takes 10 numbers from user; store them in one dimension array. Display the sum of all even numbers. Also display the sum of all odd numbers
- 6) Write a program to sort an array of 10 integers. Use bubble sort.
- 7) Write a program to sort an array of 10 integers. Use selection sort.
- 8) Write a program to rotate elements of array in given direction (left or right) by n places. Assume array to be circular.
- 9) Write a program to implement stack using array.
- 10) Write a program to find duplicate entry in an array of size 10.
- 11) Write a program to fill an array of size 10 with integers from 0 to 9 in random order. Sequence of numbers must be random every time you run the program.
- 12) Write a program to merge two arrays into the third one. Take two sorted arrays of different sizes.
- 13) Write a program to swap two arrays.
- 14) Write a program to calculate the frequency of elements in an array of size 20.

#11 Arrays

C Notes Vol-2 by Saurabh Shukla

www.mysirg.com
