

Arrays and Strings in C Language

Module 01 - One dimensional arrays in C Language

 CBSE

 ICSE

 NTSE

 Banking & Insurance

 Central Govt. Service

 State Govt. Services

 LAW Entrance

 MBA Entrance

 Railways & Metro Services

...many more

abhyasonline.in

Introduction:

A data structure is the way data is stored in the machine and the functions used to access that data. An easy way to think of a data structure is a collection of related data items. An array is a data structure that is a collection of variables of one type that are accessed through a common name.

- Each element of an array is given a number by which we can access that element which is called an index. It solves the problem of storing a large number of values and manipulating them.
- Arrays Previously we use variables to store the values. To use the variables we have to declare the variable and initialize the variable i.e, assign the value to the variable.
- Suppose there are 1000 variables are present, so it is a tedious process to declare and initialize each and every variable and also to handle 1000 variables.
- To overcome this situation we use the concept of array .In an Array values of same type are stored. An array is a group of memory locations related by the fact that they all have the same name and same type. To refer to a particular location or element in the array we specify the name to the array and position number of particular element in the array.

Definition:

An array in C is a collection of elements of the same data type stored in contiguous memory locations. Arrays allow you to store multiple values in a single variable, making it easier to manage and manipulate data efficiently.

Key Features of Arrays:

1. **Fixed Size:** The size of an array is defined at the time of declaration and cannot be changed later.
2. **Homogeneous Elements:** All elements in an array must be of the same data type (e.g., int, float, char).
3. **Contiguous Memory:** Array elements are stored in adjacent memory locations.
4. **Indexing:** Array elements are accessed using indices, starting from 0.

Course
&
Test Series

 CBSE

 ICSE

 NTSE

 Banking &
Insurance

 Central Govt.
Service

 State Govt.
Services

 LAW
Entrance

 MBA
Entrance

 Railways & Metro
Services

...many more

abhyasonline.in

Introduction to 'C' Language

Declaring and Initializing Arrays

Syntax:

```
c  
  
data_type array_name[array_size];
```

 Copy code


Example:

```
c  
  
int numbers[5]; // Declares an array of 5 integers
```

 Copy code

You can also initialize an array at the time of declaration:

```
c  
  
int numbers[5] = {1, 2, 3, 4, 5}; // Initializes the array with 5 values
```

 Copy code

If the size is omitted, the compiler determines it from the number of initial values:

```
c  
  
int numbers[] = {1, 2, 3, 4, 5};
```

 Copy code

Accessing Array Elements


Array elements can be accessed using their index:

```
c  
  
int value = numbers[2]; // Access the 3rd element (index 2)
```

 Copy code

You can modify array elements similarly:

```
c  
  
numbers[1] = 10; // Sets the 2nd element (index 1) to 10
```

 Copy code

Course
&
Test Series

Introduction to 'C' Language

One-Dimensional Array in C

A one-dimensional (1D) array in C is a simple linear structure that stores multiple elements of the same data type in contiguous memory locations. It allows for efficient storage and manipulation of a collection of data using a single variable name.

Declaring a 1D Array

Syntax:

```
c  
data_type array_name[array_size];
```

- `data_type`: Specifies the type of elements (e.g. `int`, `float`, `char`).
- `array_name`: The name of the array.
- `array_size`: Number of elements the array can hold.

Example:

```
c  
int numbers[5]; // Declares an array of 5 integers
```

Initializing a 1D Array

Method 1: During Declaration

```
c  
int numbers[5] = {1, 2, 3, 4, 5};
```

Method 2: Partial Initialization

```
c  
int numbers[5] = {1, 2}; // Remaining elements are set to 0
```

Method 3: Implicit Size

```
c  
int numbers[] = {10, 20, 30}; // Compiler automatically determines size (3 elements)
```

 CBSE

 ICSE

 NTSE

 Banking & Insurance

 Central Govt. Service

 State Govt. Services

 LAW Entrance

 MBA Entrance

 Railways & Metro Services

...many more

abhyasonline.in

Course
&
Test Series

 CBSE

 ICSE

 NTSE

 Banking &
Insurance

 Central Govt.
Service

 State Govt.
Services

 LAW
Entrance

 MBA
Entrance

 Railways & Metro
Services

...many more

abhyasonline.in

Introduction to 'C' Language

Accessing and Modifying Elements

Array elements are accessed using index values, starting from 0.

Accessing:

```
c Copy code  
  
int first = numbers[0]; // Accesses the first element
```

Modifying:

```
c Copy code  
  
numbers[1] = 25; // Changes the second element to 25
```

Example: Print All Elements

```
c Copy code  
  
#include <stdio.h>  
  
int main() {  
    int numbers[] = {10, 20, 30, 40, 50};  
  
    for (int i = 0; i < 5; i++) {  
        printf("Element at index %d: %d\n", i, numbers[i]);  
    }  
  
    return 0;  
}
```

Advantages of 1D Arrays

1. **Simplifies Data Management:** Store multiple related values under one variable name.
2. **Efficient Access:** Elements can be accessed in constant time using indices.
3. **Compact Code:** No need to declare separate variables for each value.

Limitations of 1D Arrays

1. **Fixed Size:** Cannot adjust the size dynamically.
2. **Sequential Access:** Searching through large arrays is time-consuming without additional optimizations.
3. **Lack of Bounds Checking:** Accessing elements beyond the array size leads to undefined behavior.

Course
&
Test Series

Introduction to 'C' Language

Solved Example: To find the sum of elements in an array

Program Code:

```
c Copy code  
  
#include <stdio.h>  
  
int main() {  
    int arr[5]; // Declare an array of size 5  
    int sum = 0; // Variable to store the sum  
  
    // Input elements into the array  
    printf("Enter 5 elements:\n");  
    for (int i = 0; i < 5; i++) {  
        scanf("%d", &arr[i]);  
    }  
  
    // Calculate the sum of the elements  
    for (int i = 0; i < 5; i++) {  
        sum += arr[i];  
    }  
  
    // Display the sum  
    printf("The sum of the elements is: %d\n", sum);  
  
    return 0;  
}
```

Assignment

Ques 1: Maximum and Minimum Element

Write a program to find the largest and smallest numbers in a 1D array.

- Input: {23, 45, 12, 67, 34}
- Expected Output: Max = 67, Min = 12

Ques 2: Count Even and Odd Elements

Create a program that counts the number of even and odd elements in an array.

- Input: {10, 15, 20, 25, 30}
- Expected Output: Even = 3, Odd = 2

...many more

abhyasonline.in

Course
&
Test Series

Introduction to 'C' Language

Ques 3: Create a program that takes 10 integers as input and counts how many of them are positive, negative, or zero.

Ques 4: Write a Program to Multiply Each Element of an Array by a Given Number

 CBSE

 ICSE

 NTSE

 Banking & Insurance

 Central Govt. Service

 State Govt. Services

 LAW Entrance

 MBA Entrance

 Railways & Metro Services

...many more

abhyasonline.in

