C_49 ⇒ Arrays in C - Part 4

Array Program 1

Arrays Notes

* Array is a collection of more than one data item of same type.

* All the data items are stored in contiguous memory location.

* Number of data items array holds is size of array.

* Once size has declared, it can't be changed at run time (fixed size).

* Index starts from 0.

* Known as derived data type

* Accessing of any element is faster using Index of array.

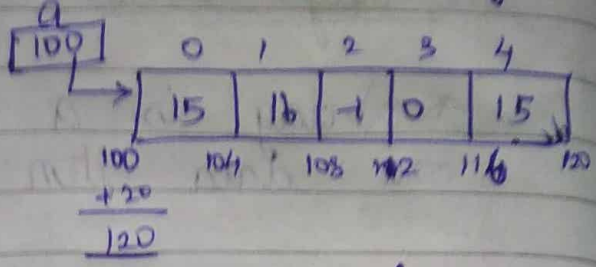
* It allows to store data in multidimensional form.

* Inserting & deleting elements from array is costly(or) tough.

(X) * No bound checking in C.
↳ Eg: a[10]; index 0 - 9
but when we access a[10] → it will give garbage value, (ie) it has no boundary checking.

Array Program ①

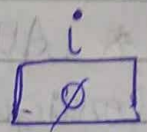
a[5] × 4 => 20 bytes




```
#include <stdio.h>
#include <conio.h>
void main()
{
  int i,a[5];
  for(i=0;i<5;i++)
  {
    scanf("%d",&a[i]);
  }
  for(i=0;i<5;i++)
  {
    printf("%d",a[i])
  }
}
```

Array element at index %d is: %d ,a[i]

i



x 1 2 3 4 5

a[0]

[B + i x sizeof(int)]

fetching array element.

=> 100 + 0x4

a[0] => 100

Array Program ②

* To print reverse elements of array.

```
void main()
{
  int i, a[5];
  for(i=0;i<5;i++)
  {
    scanf("%d",&a[i]);
  }
  for(i=4;i>=0;i--)
  {
    printf("%d\n",a[i]);
  }
}
```

a[3] =>

100 + 3x4

100 + 12

a[3] = 112

CODE 1:

```c
#include <stdio.h>
#include <stdlib.h>
/** 1 - ARRAY PROGRAM **/
/** PRINT ARRAY ELEMENTS IN BOTH FORWARD AND REVERSE ORDER **/
int main()
{
    int i,a[5];
    printf("Enter the array elements:\n");
    for(i=0;i<5;i++)
     scanf("%d",&a[i]);
     printf("/** Forward order **/\n");
    for(i=0;i<5;i++) /** Forward order **/
    printf("The array element at index %d is:%d\n",i,a[i]);
    printf("/** Reverse order **/\n");
    for(i=4;i>=0;i--) /** Reverse order **/
    printf("The array element at index %d is:%d\n",i,a[i]);
    getch();
}
```

```
"D:\1. C NOTEBOOK\C LANGUAGE\C PROGRAMS\PART 5_Jennys
Enter the array elements:
1
2
3
4
5
/** Forward order **/
The array element at index 0 is:1
The array element at index 1 is:2
The array element at index 2 is:3
The array element at index 3 is:4
The array element at index 4 is:5
/** Reverse order **/
The array element at index 4 is:5
The array element at index 3 is:4
The array element at index 2 is:3
The array element at index 1 is:2
The array element at index 0 is:1
```