

Chapter 2

Arrays

An array is a finite collection of homogeneous elements. In an array elements are stored in contiguous memory location. Each element in an array is accessed by an index (Subscript).

Note: A subscript or index is a number which indicates the position of an element in an array.

Declaring an Array

An array is declared as

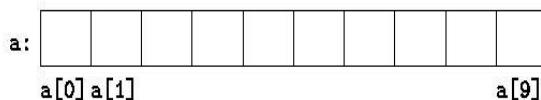
Data type arrayname[Size];

The total number of elements in array is called Size of the array. In an array the index or subscript starts at 0 is called base index.

For Example

`int a[10];`

Declares an array of size 10, where the first element is at `a[0]` and the last element is at `a[9]`.



The memory space allocated to an array can be calculated as

*Total bytes = size of array * size of data type*

Or

*Total bytes = size of (data type) * number of array elements*

For example `int a[10]`, the total bytes will be $2 \times 10 = 20$ Bytes. (Since each element takes 2 Bytes).

For example `float b[20]`, the total bytes = $4 \times 20 = 80$ Bytes (Size of Float data type is 4 Bytes)

Initializing an array

Arrays can be initialized at the time of declaration like variables. The following statements show how an array can be declared.

`int a[4] = {8, 7, 2, 4, 9};`

here a[0] will be 8,a[1] will be 7 and so on.

Examples

```
Char ch[4]={'A','e','i','O','U'};
```

```
char greeting[] = "Hello";
```

```
char greeting[] = {'H','e','l','l','o','\0'};
```

string "Hello" in a memory:

'H'	'e'	'l'	'l'	'o'	'\0'
-----	-----	-----	-----	-----	------

Every String in C++ is terminated by a Null Character.

Note:C++ does not have string data type rather a string is implemented as a single dimension character array.

Accessing elements of an array

Once an array is declared,elements are referenced by an index or subscript.The index starts at 0.

Entering data to an array

The for loop can be used to enter data to an array

```
For(i=0;i<=9;i++)
```

```
{
```

```
    cin>>a[i];
```

```
}
```

Program to initialize an array and print the elements

```
#include<iostream.h>
```

```
#include<conio.h>
```

```
void main()
```

```
{
```

```
    Int a[5]={10,9,8,7,6,5};
```

```
cout<<"The content of array "<<"\n";

for(i=0;i<=5;i++)

{

    cout<<i;

}

}
```

Output:

10 9 8 7 6 5

Program to input elements to an array and print it.

```
#include<iostream>

using namespace std;

int main(void)

{

    int a[100],n,i;

    cout<<"Enter the Limit";

    cin>>n;

    cout<<"Enter the elements";

    for(i=0;i<=n;i++)

    {

        cin>>a[i];

    }

    cout<<"Elements Of the Array are \n";

    for(i=0;i<=n;i++)

    {
```

```
cout<<a[i];  
cout<<"\n";  
}  
}
```

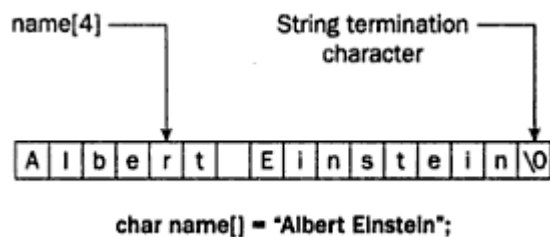
Program to read n elements to an array and find its sum.

```
#include<iostream>  
  
using namespace std;  
  
int main(void)  
{  
    int a[100],n,i,s=0;  
    cout<<"Enter the Limit";  
    cin>>n;  
    cout<<"Enter the elements";  
    for(i=0;i<=n;i++)  
    {  
        cin>>a[i];  
    }  
  
    for(i=0;i<=n;i++)  
    {  
        s=s+a[i];  
    }  
    cout<<"Sum ="<<s;  
}
```

String handling using array

A string is an array of characters terminated by a null character(\0).A string can be declared as

```
char name[ ]="Albert Einstein";
```



Note:Each character in a string occupies one byte.

Another way for declaring a string is

```
char name[6]="Smith";
```

```
char name[6]={'S', 'm', 'i', 't', 'h', '\0'};
```

Program to read your name using array

```
#include<iostream>

using namespace std;

int main(void)
{
    char Name[10];

    cout<<"Enter the name";

    cin>>Name;

    cout<<"How are you " <<Name;

}
```

Output:

Enter the name

Smith

How are you Smith

gets () function

In inputting strings in C++ a white space is treated as data separator. The gets () function is used to read a string through the keyboard including whitespace. The syntax is

gets(character array_name);

```
#include<iostream>

#include<cstdio>          //header file for gets ( ) function

using namespace std;

int main(void)
{
    char Name[10];
    cout<<"Enter the name";
    gets(Name);
    cout<<"How are you " <<Name;
}
```

puts () function

The puts () function is used to display a string on the screen. The syntax is

puts(string)

Example: A program to read a name using gets () function and display it using puts () function.

```
#include<iostream>

#include<cstdio>

using namespace std;
```

```
int main(void)
{
    char Name[10];
    cout<<"Enter the name";
    gets(Name);
    cout<<"The Entered Name is"<<"\n";
    puts(Name);
}
```