Array :- 1

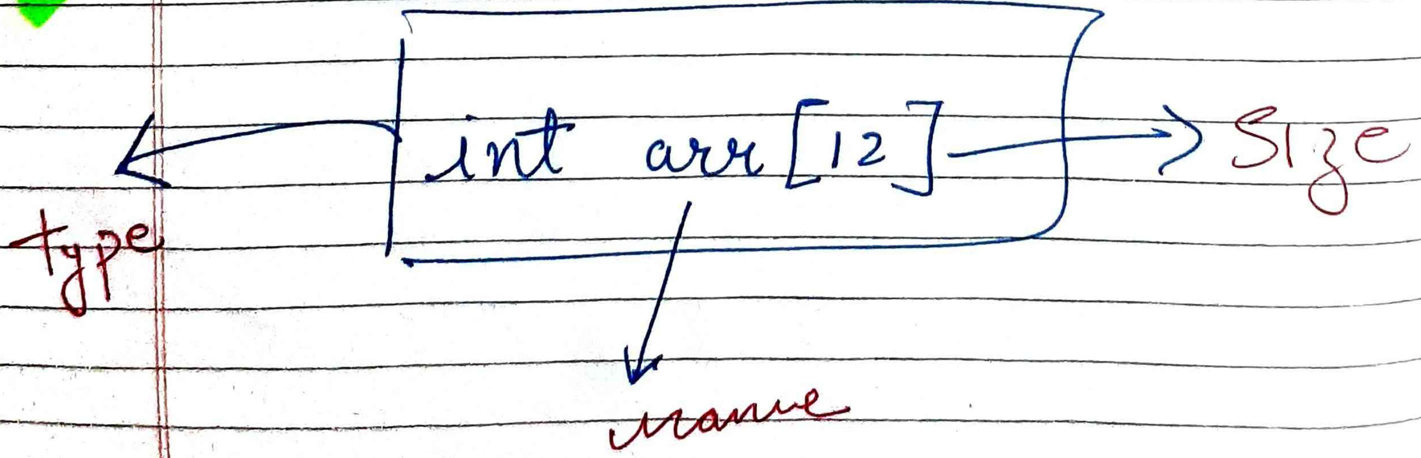
① What is array →

- → Collection of elements
- → Similar elements
- → list of Similar items
- → data structure

② Why We need array

- → Need to create large set of data under a single Variable name.

③ Creating array →



```
int arr[12]
```

Note →

arr and &arr will print the base address of array

④ Initialization

```
int arr[] = {2, 3, 4, 5, 6, 7, 8}
```

*

```
int arr [10] = {2, 3, 4, 5, 6}
```

↓

The array contains 5 elements but can store 10 so rest elemets are

{2, 3, 4, 5, 6, 0, 0, 0, 0, 0}

This type of array is static one fixed cannot be changed.

Note → To create dynamic array we use Vector.

Note →

```
int n;
Cin >> n;
int arr [n];
```

} Bad practice

Note → There is a limit on creating array.

(5) Index & & Access in Array

```
int arr[5] = {10, 20, 30, 40, 50}
```

	104	108	112	116	120
	10	20	30	40	50
Index →	0	1	2	3	4

5 size array

index [0 → 4]

n size array → index [0 → (n-1)]

6 Why Index of Array start from 0

$\boxed{arr[0]}$

$arr + 0 \times \underbrace{4}$ → size of integer

→ Value at this address.

example

arr[1] → Value at (arr + 1 x 4)

Value at (104 + 1x4)

Value at (108)

memset() function

printing array

```
   1       2           4
for (i=0; i<5; i++){
              3
                          i = 0
                          i = 1
}                         i = 2
                          i = 3
                          i = 4
```

7 Array & & functions:

```
int main(){
    int arr[] = {5,6};
    int (arr)
    print arry().
}
```

```
inc (int arr[])
{
   arr[0]= arr[0]+10
   print arry()
}
```

Example

```
main ()
{
   int a[]={1,2,3}
   function2(a,size)
   print (a)
}.
```

```
function2(a[], size)
{
   a[0] =20;
   a[1] =30;
   a[2] =40;
   printArry ()
}
```

Note ÷

▲ Always pass the size of array when you pass array to function

Example

With size of() we will get the overall size but not the no of element present.

→ This is only because if an array with size 10 has 3 element then it is not possible to preidict the element

```
int arr[10] = {1, 2, 3}
```

↓

rest 7 element are initialized with 0

① pass by Value → The Value is not changed

② pass by Reference → The Value is changed in array.

⑧ Linear Search ÷

2	9	6	7	4	12	15

```
for (int i=0; i< Size; i++)
   {
      if (arr[i] = = Target)
            return i;
   }
            return -1;
}
```

target → Target of element to find

Size → Size of array.

Question to perform :-

1. Count 0 and 1 in an array
2. Maximum array in array
3. Minimum array in array
4. Extreme print in Array:
5. Reverse an array
6. Swap function implementation