

ARRAYS

WHAT IS AN ARRAY

- **An array is a derived data type (derived from fundamental data type)**
- **It is a collection of variables of the same type that are referenced by a common name.**
- **Consist of contiguous memory locations.**
- **Lowest address corresponds to first element**
- **Highest address corresponds to the last element.**
- **Can have data items of type like: int, char, float and also user-defined types like : structures, objects.**

NEED FOR AN ARRAY

- To store large number of variables of same type under a single variable.
- Easy understanding of the program.
- E.g.
 - To store Marks of 50 students.
 - Record of sales of 100 salesman.

TYPES OF ARRAYS

- **Single Dimensional Array:** Element specified by single subscript
- **Syntax:**

`type array_name [size]`

↑
Base
type of
array

↑
Name
of array

↑
No. of elements that can
be stored:

Can be a integer value
without the sign

Creating an Array

```
void main( )
{
    int a[10]; // declaration of an array 'a'
    int n;

    // input 10 elements in an array

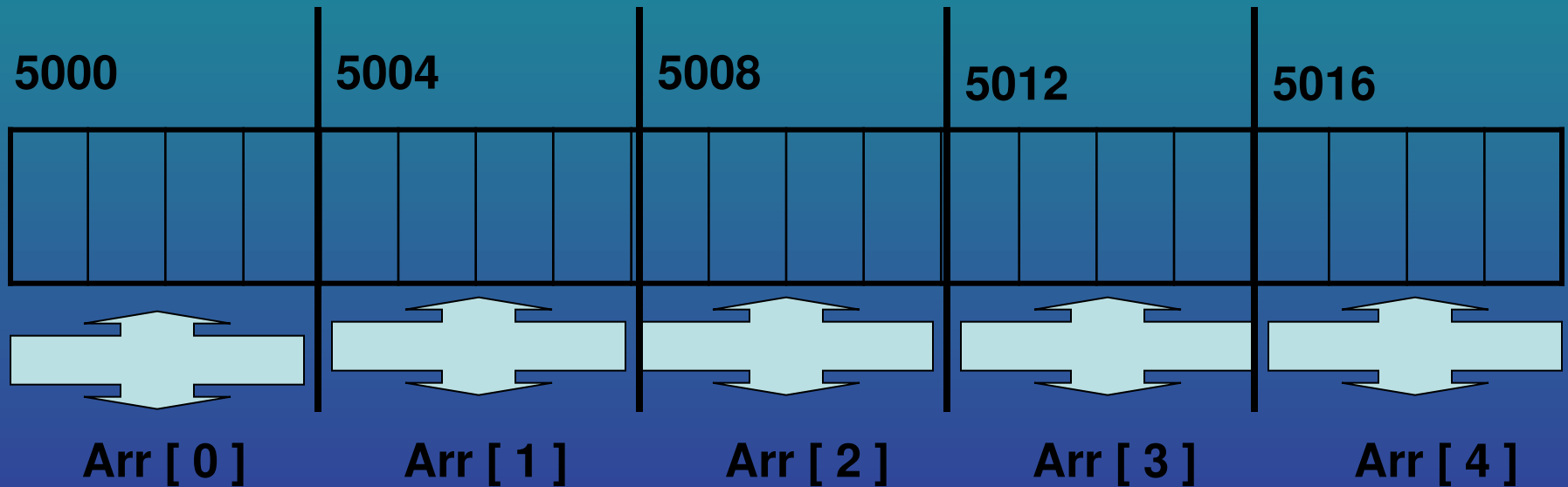
    for ( n = 0; n < 10 ; n + + )
    {
        cin >> a [ n ];
    }

    // display the 10 elements of the array input

    for ( n = 0 ; n < 10 ; n + + )
    {
        cout << a [ n ] << endl;
    }
}
```

Memory Representation of Single Dimension Array

- If the array is float arr [5];
memory representation would be as follows:



Total Memory requirement is : size of (type) * size of array

$$4 * 5 = 20 \text{ bytes}$$

ARRAY INITIALISATION

```
int list [ 5 ] ; // declaration
```

```
int list [ 5 ] = { 10, 20, 30, 40, 50 } ;  
// declaration & initialization
```

UNSIZEED ARRAY INITIALISATION

- Can skip the size of an array in array initialization
- Elements of an array can be added or removed without changing array dimensions.

E.g.

```
float price [ ] = { 50.5, 63.97, 84.6, 779.8 };
```


Program to count the no. of employees earning more than Rs. 1 lakh per annum. Monthly salaries of 10 employees are given.

```
void main ( )
{
    const int size = 10 ;
    float sal [ size ] , an_sal ;
    int count = 0;

    // loop to accept monthly salaries of 10 employees

    for ( int j = 0 ; j < size ; j + + )
    {
        cout << " Enter the monthly salary of employee " << j + 1 ;
        cin >> sal [ j ];
    }
}
```

```
// loop to count employees earning more than Rs. 1 lakh per annum
for ( j = 0 ; j < size ; j + + )
{
    an_sal = sal [ j ] * 12 ;
    if ( an_sal > 100000 )
    {
        count ++ ;
    }
}

cout << count << “ employees out of “ << size << “
employees are earning more than Rs. 1 lakh per annum “ ;
}
```

WAP to input 10 numbers in an array and replace all even no.s by 0 and odd no.s by 1

```
void main ( )
{
    int a [ 10 ], n;
    // loop to accept 10 values in an array 'a'
    for ( n = 0; n < 10 ; n + + )
    {
        cin >> a [ n ];
    }
    // loop to check if the element of an array is even replace by 0
    // and if odd replace by 1
    for ( n = 0; n < 10 ; n + + )
    {
        if ( ( a [ n ] % 2 ) == 0 )
        {
            a [ n ] = 0;
        }
        else
        {
            a [ n ] = 1 ;
        }
    }
}
```

```
// display the 10 elements of the  
array
```

```
for ( i = 0 ; i < 10 ; i + + )  
{  
    cout << a [ i ] << endl;  
}
```

WAP to find the largest and smallest no. in an array of 10 elements

```
// input an array
// display the array
// to find the largest element
int largest = a [ 0 ] ;
for ( int i = 1 ; i < 10 ; i + + )
{
    if ( a [ i ] > largest )
    {
        largest = a [ i ] ;
    }
}
cout << " largest value is : " << largest ;
```

WAP to find the largest and smallest no. in an array of 10 elements

```
// input an array
// display the array
// to find the lowest element
int lowest = a [ 0 ];
for ( n = 1 ; n < 10 ; n + + )
{
    if ( a [ n ] < lowest )
    {
        lowest = a [ n ];
    }
}
cout << " lowest value is : " << lowest ;
```