ARRAYS

MODULE 3

- A fixed size sequenced collection of elements of the same data type.
- A collection of variables of the same data type that are referenced by a common name.

```
int rollno[64];
```

Different types of Array

- One Dimensional Arrays
- Two Dimensional Arrays

One Dimensional Arrays

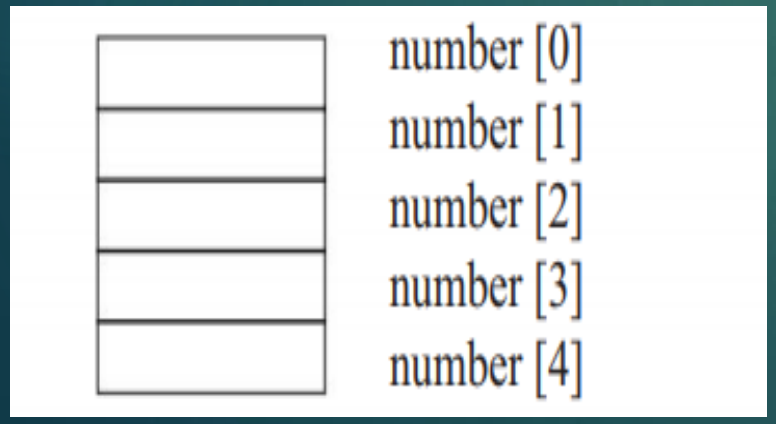
- A list of items can be given one variable name using only one subscript and such a variable is called a single subscripted variable or a one dimensional array.

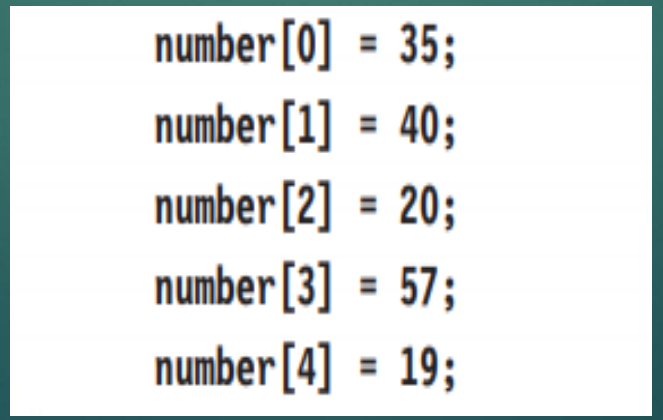
$$A = \frac{\sum_{i=1}^{n} x_i}{n}$$

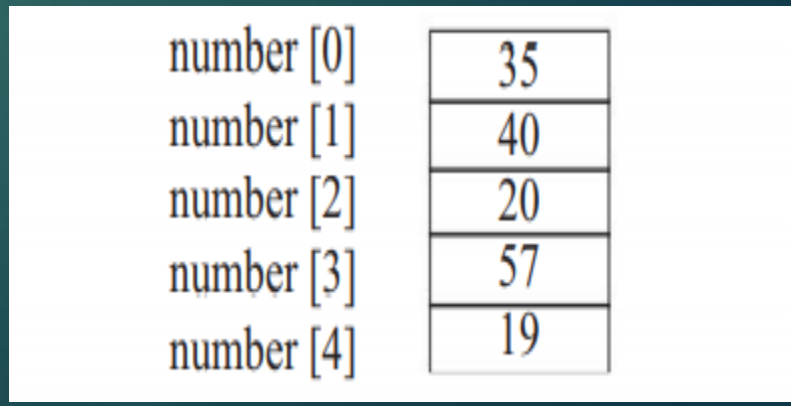
- Example: To calculate the average of n values of x.

- Set of five numbers (35,40,20,57,19) by any array variable number.
- Declare the variable number as

```
int number[5];
```







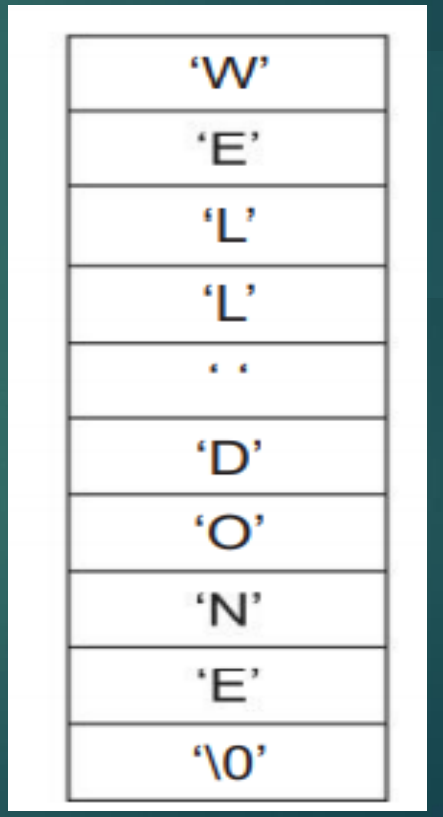
Declaration of One Dimensional Arrays

Data type variable-name[size];

Example : int group[10];

char name[10];

"WELL DONE"



Simple Program using Array

```
#include<stdio.h>
void main()
{
int avg,sum=0;
int i;
int marks[50];
for (i=0;i<=49;i++)
{
printf("enter marks:");
scanf("%d", &marks[i]);
}
for (i=0;i<=49;i++)
{
Sum = sum + marks[i];
Avg = sum/50;
printf("Average marks = %d", avg);
}
```

Initialization of array

- At compile time
- At run time

Compile time Initialization

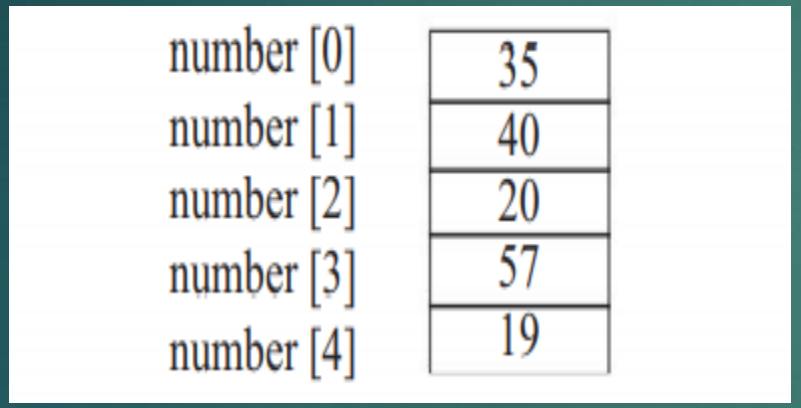
Data type array-name[size] = {list of variables};

Run Time Initialization

```
for (i=0;i<100;i++)
{
if i< 50
  a[i] = 0;
else
  a[i] =1;
}
```

Memory allocation and Accessing of Array

Base address = 2000



- Access first element a[0] = 35
- a[3]= 57

To access any element of the array at any time :

Base address + index * size of the data type

- To read 5 elements in an array and print the values.
- To read 5 elements and print the elements in reverse order.
- To find the sum and average of 5 marks using the concept of arrays.

Linear Search

```c
#include <stdio.h>
void main()
{
 int array[100], search, c, n,found;
 printf("Enter number of elements in array\n");
 scanf("%d", &n);
 printf("Enter %d integer(s)\n", n);
 for (c = 0; c < n; c++)
 scanf("%d", &array[c]);
 printf("Enter a number to search\n");
 scanf("%d", &search);
 for (c = 0; c < n; c++)
 {
 if (array[c] == search) /* If required element is found */
 {
 found=1;
 break;
 }
 }
 if (found == 1)
 printf("%d is present at location %d.\n", search, c+1);
 else
 printf("%d isn't present in the array.\n", search);
 }
```