ARRAYS

Session Objectives

- Explain Arrays
- Explain Declaration,Initialization of Array
- Explain Types of Array
- One Dimensional,Two Dimensional and Multi Dimensional Array

Array is defined as a set of homogeneous data items.

An Array is a group of elements that share a common name that are differentiated from one another by their positions within the array

DECLARATION OF AN ARRAY

Datatype arrayname[subscript];

POINTS TO BE NOTED :

1) **Arrayname should be a valid “C” variable**
2) **Arrayname should be unique**
3) **The elements in the array should be of same type**
4) **Subscript (array size) cannot be negative**
5) **Subscript must always be an integer**

TYPES OF ARRAY

One Dimensional Array

Two Dimensional Array

Multi Dimensional Array

Single or One Dimensional Arrays

- **Arrays whose elements are specified by one subscript are called One dimensional array or linear array.**

- **Syntax :**

datatype arrayname[size];

- ***For Example :***

int a [3]

- ***Note :***

By default array index should starts with zero (0)

Write a program for entering data into an array & Reading data from an array

```
#include<stdio.h>
void main()
{
int arr[10],I,n;
printf("\n ENter N Elements");
scanf("%d",&n);
for(i=0;i<n;i++)
{
printf("enter arr[%d]=",i);
scanf("%d",&arr[i]);
}
for(i=0;i<n;i++)
{
printf("%d\n",arr[i]);
}
}
```

Input

```
Enter N Elements   : 3
Enter arr[0]  :  2
Enter arr[1]  :  5
Enter arr[2]  :  3
```

Output

```
2
5
3
```

PROGRAM-ARRAY INITIALIZATION

Array Initialization

```
#include<stdio.h>
#include<conio.h>
void main()
{
int a[5]={10,20,30,40,50};
int i;
clrscr();
for(i=0;i<5;i++)
{
printf("%d\n",a[i]);
}
getch();
}
```

Output

```
10
20
30
40
50
```

Write a “C” program to sort the given number is in ascending order using one dimensional array

```
#include<stdio.h>
void main()
{
int i,j,n, a[10],temp;
printf("\n size of vector=");
scanf("%d",&n);
printf("vector elements:");
for (i=0;i<n;i++)
{
scanf("%d",&a[i]);
}
for(i=0;i<n-1;i++)
for(j=i+1;j<n;j++)
if(a[i]>a[j])
{
temp=a[i];
a[i]=a[j];
a[j]=temp;
}
printf("\n\nElements in asending order is=\n");
for(i=0;i<n;i++)
printf("%d",a[i]);
printf("\n\nElements in descending order is=\n");
for(i=n-1;i>=0;i--)
printf("%d",a[i]);
getch();
}
```

Two Dimensional Arrays

- A **Arrays whose elements are specified by two subscript such as row and column are called One dimensional array or linear array.**
- **Row → means horizontally**
- **Column → means vertically**

- **A two - dimensional array looks like a school time-table consisting of rows and columns.**

- **A two – dimensional array is declared as -**

int a [3] [3]

Two Dimensional Array Initialization

```
int ary [3] [4] = { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12 };
```

- The result of the above assignment will be as follows :

ary [0] [0] = 1	ary [0] [1] = 2	ary [0] [2] = 3	ary [0] [3] = 4
ary[1] [0] = 5	ary [1][1] = 6	ary [1] [2] = 7	ary [1][3] = 8
ary[2] [0] = 9	ary [2][1] = 10	ary [2] [2] = 11	ary [2][3] = 12

(OR)

```
int ary [3] [4] =
        {
        { 1, 2, 3 },
        { 4, 5, 6 },
        { 7, 8, 9 },
      { 10, 11, 12 }
        };
```

Write a “C” program to perform the addition of two matrices

```
#include<stdio.h>
void main()
{
int a[3][3],b[3][3],c[3][3],i,j;
printf("Input A - Matrix\n");
for(i=0;i<3;i++)
for(j=0;j<3;j++)
scanf("%d",&a[i][j]);
printf("Input B - Matrix\n");
for(i=0;i<3;i++)
for(j=0;j<3;j++)
scanf("%d",&b[i][j]);
for(i=0;i<3;i++)
for(j=0;j<3;j++)
c[i][j]=a[i][j]+b[i][j];
printf("Sum of A and B Matrix=\n");
for(i=0;i<3;++i)
{
for(j=0;j<3;++j)
printf("%d",c[i][j]);
printf("\n");
}
}
```

Write a “C” program to perform the subtraction of two matrices

```
#include<stdio.h>
void main()
{
int a[3][3],b[3][3],c[3][3],i,j;
printf("Input A - Matrix\n");
for(i=0;i<3;i++)
for(j=0;j<3;j++)
scanf("%d",&a[i][j]);
printf("Input B - Matrix\n");
for(i=0;i<3;i++)
for(j=0;j<3;j++)
scanf("%d",&b[i][j]);
for(i=0;i<3;i++)
for(j=0;j<3;j++)
c[i][j]=a[i][j]-b[i][j];
printf("Sum of A and B Matrix=\n");
for(i=0;i<3;++i)
{
for(j=0;j<3;++j)
printf("%d",c[i][j]);
printf("\n");
}
}
```

Write a “C” program to sort the given names in Alphabetical order using One dimensional array

```
#include<stdio.h>
#include<string.h>
void main()
{
int i,j,n;
char a[10][10],temp[10];
printf("\n Enter the N Values");
scanf("%d",&n);
printf("Enter the Names one by one :\n");
for(i=0;i<n;i++)
{
scanf("%s",&a[i]);
}
for(i=0;i<n-1;i++)
for(j=i+1;j<n;j++)
if((strcmp(a[i],a[j]))>0)
{
strcpy(temp,a[i]);
strcpy(a[i],a[j]);
strcpy(a[j],temp);
}
printf("The Names in Alphabetical Order is =\n");
for(i=0;i<n;i++)
printf("\n%s",a[i]);
}
```

Write a “C” program to perform matrix multiplication using two dimensional array

```
#include<stdio.h>
void main()
{
int a[10][10],b[10][10],c[10][10],i,j,m,n,p,q,k;
printf("Input row and column of A matrix \n");
scanf("%d %d",&n,&m);
printf(" Input row and column of B matrix \n");
scanf("%d %d",&p,&q);
if(n==q){
printf(" Matrices can be Multiplied: \n");
printf(" Input A-matrix \n");
for(i=0;i<n;++i)
for(j=0;j<m;++j)
scanf("%d",&a[i][j]);
printf(" Input B-matrix \n");
for(i=0;i<p;++i)
for(j=0;j<q;++j)
scanf("%d",&b[i][j]);
printf("The resultant matrix is\t:\n");
for(i=0;i<n;++i){
for(j=0;j<m;++j){
c[i][j]=0;
for(k=0;k<m;++k)
c[i][j]=c[i][j]+a[i][k]*b[k][j];
printf("%d",c[i][j]);}
printf("\n");}}
else
printf("Matrices cannot be multiplied \n");
}
```

Write a “C” program to find the largest and smallest numbers given in the array

```
#include<stdio.h>
void main()
{
int i,n;
float a[20],large,small;
printf("\nEnter the N  values=");
scanf("%d",&n);
printf("Enter the values one by one :\n");
for(i=0;i<n;i++)
{
scanf("%f",&a[i]);
}
large=a[0];
for(i=1;i<n;i++)
{
if(a[i]>large)
large=a[i];
}
small=a[0];
for(i=1;i<n;i++)
{
if(a[i]<small)
small=a[i];
}
printf("Largest element is = %f\n",large);
printf("Smallest element  = %f\n",small);
}
```

Session Summary

- Arrayname should be a unique and valid “C” Variable name
- The number of elements in a multi dimensional array is the product of its subscripts
- Arrays can be initialized to the same type in which they are declared
- The character array receives the terminating ‘\0’ in the string constant
- The individual values in the array are called as elements
- It is not necessary to specify the length of an array, explicitly in case if initializers are provided for the array during declaration itself

EXERCISES

1. Write a program to search an element and to find how many times it is present in the array?
2. Write a program to find the sum of diagonal elements in a matrix
3. Write a program to find the second largest number in an array?
4. Write a program to remove the duplicate elements of the array?
5. Write a program to merge two arrays and print the merged array in ascending order?
6. Write a program to insert an element into an sorted array of integers?
7. Write a program to display only the negative elements of the array?