Unit 6
Arrays

Introduction to Array

- Arrays are used to store multiple values in a single variable, instead of declaring separate variables for each value.
- An array is defined as the collection of similar type of data items stored at contiguous memory locations.
- Arrays are the derived data type in C programming language which can store the primitive type of data such as int, char, double, float, etc.
- It also has the capability to store the collection of derived data types, such as pointers, structure, etc.
- The array is the simplest data structure where each data element can be randomly accessed by using its index number.

6.2 Types of Array

- There are mainly three types of the array:
  1. One Dimensional (1D) Array
  2. Two Dimension (2D) Array
  3. Multidimensional Array

One Dimensional (1D) Array

Declaration of Array

```
data_type array_name[array_size];
```

Example:

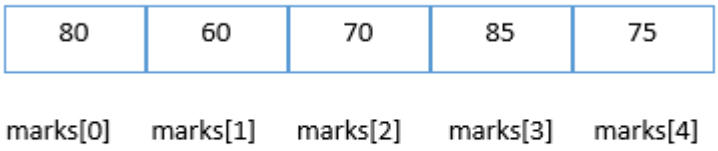
```
int marks[5];
```

Here, int is the data_type, marks are the array_name, and 5 is the array_size.

Initialization of Array

- The simplest way to initialize an array is by using the index of each element.
- Consider the following example.

```
marks[0]=80;          //initialization of array
marks[1]=60;
marks[2]=70;
marks[3]=85;
marks[4]=75;
```



Initialization of Array

Declaration with Initialization of Array

Example:

```
int marks[5]={20,30,40,50,60};
```

Or

```
int marks[]={20,30,40,50,60};
```

Accessing Array Elements

- An element is accessed by indexing the array name.
- This is done by placing the index of the element within square brackets after the name of the array.

Example:

```
double salary = balance[9];
```

Ex1: Write a simple program using array.

Ex2: Write a program to sort element of an array.

Ex3: Write a program to find largest and second largest element of an array.

Two Dimensional (2D) Array

Declaration of 2D Array

```
data_type array_name[row_size] [column_size];
```

Example:

```
int arr[3][4];
```

Initialization of 2D Array

- Multidimensional arrays may be initialized by specifying bracketed values for each row.
- Following is an array with 3 rows and each row has 4 columns.

```
int a[3][4] = {
```

```
  {0, 1, 2, 3} ,   /*  initializers for row indexed by 0 */
  {4, 5, 6, 7} ,   /*  initializers for row indexed by 1 */
  {8, 9, 10, 11}   /*  initializers for row indexed by 2 */
};
```

OR

```
int a[3][4] = {0,1,2,3,4,5,6,7,8,9,10,11};
```

Accessing Array Elements

- An element in a two-dimensional array is accessed by using the subscripts, i.e., row index and column index of the array.

Example:

```
int val = a[2][3];
```

Example: Program to demonstrate the concept of 2D array.

```
#include <stdio.h>
int main()
{
        int i, j;
        int arr[2][3] = { 10, 20, 30, 40, 50, 60};
        for(i = 0; i < 2; i++)
        {
                for(j = 0; j < 3; j++)
                {
                        printf("%d\t", arr[i][j]);
                }
                printf("\n");
        }
        return 0;
}
```

Output:

```
10    20    30
40    50    60
```

Character Array and Strings

- String is a sequence of characters that are treated as a single data item and terminated by a null character '\0'.

Syntax:

```
char variable_name[size] = {list of string};
```

Example 1:

```
#include <stdio.h>
int main ()
{
        char str1[6] = {'H', 'e', 'l', 'l', 'o', '\0'};
        printf("Greeting message: %s\n", str1 );

        char str2[6] = "Hello";
        printf("Greeting message: %s\n", str2);

        char str4[20];
        scanf("%[^\n]", &str4);
        printf("Greeting message: %s\n", str4);

        return 0;
}
```

Output:

```
Greeting message: Hello
Greeting message: Hello
Hello World
Greeting message: Hello World
```

Example 2:

```
#include <stdio.h>
int main () {
        char str[20];
        gets(str);
        printf("Greeting message: ");
        puts(str);
        return 0;
}
```

Output:

```
Hello World
Greeting message: Hello World
```

String Library Functions

- C language supports a large number of string handling functions that can be used to carry out many of the string manipulations.

- These functions are packaged in the string.h library. Hence, you must include string.h header file in your programs to use these functions.

Method	Description
strcat()	It is used to concatenate(combine) two strings
strlen()	It is used to show the length of a string
strrev()	It is used to show the reverse of a string
strcpy()	Copies one string into another
strcmp()	It is used to compare two string

Example:

```
#include <stdio.h>
#include <string.h>
int main ()
{
        char str1[12] = "Hello";
        char str2[12] = "World";
        char str3[12];
        int  len ;

        /* compare str1 and str2 */
        printf("strcmp(str1) :  %d\n", strcmp(str1, str2));

        /* copy str1 into str3 */
        strcpy(str3, str1);
        printf("strcpy( str3, str1) :  %s\n", str3 );

        /* concatenates str1 and str2 */
        strcat( str1, str2);
        printf("strcat( str1, str2):   %s\n", str1 );

        /* total lenghth of str1 after concatenation */
        len = strlen(str1);
        printf("strlen(str1) :  %d\n", len );

        /* Reverse str1 */
        printf("strrev(str1) :  %s\n", strrev(str1));

        return 0;
}
```

Output:

```
strcmp(str1) :  -1
strcpy( str3, str1) :  Hello
strcat( str1, str2):   HelloWorld
```

```
strlen(str1) :  10
strrev(str1) :  dlroWolleH
```

Exercise

1. Write a program to find sum and average of 10 integer numbers stored in an array. (5) [TU 2074]
2. Define array? What are the benefits of using array? Write a program to add two matrices using array. (10) [TU 2075]
3. Discuss any five string library functions. (5) [TU 2075]
4. What are the benefits of using arrays? Compare one dimensional array with two dimensional array. Write a program to find transpose of a matrix. (10) [TU 2077]
5. Explain any three string functions. Write a program to check if two matrices are identical or act. (5) [TU 2078]
6. Write a program to find the second largest number in the given array of numbers. (5) [TU 2079]
7. Write a program to sort element of an array.
8. "Size of character array is always declared one more than the input size." Justify the statement. Write a program to read a character array input as "TRIBHUVAN UNIVERSITY" from the user and find out how many times a character „I" occurs in that array? (1+4) [TU Model]
9. Write syntax to declare and initialize 2-dimensional array? With suitable program logic explain how would you find transpose of a 3*3 matrix? (1+4) [TU Model]