

Arrays

1-11-25

Aim – Write a program to arrange numbers in an array into descending order and another multiply two matrices (2d arrays).

Theory –

We used arrays in C – It is a collection of elements of the same data type stored in contiguous memory locations. It allows efficient data storage and access using indices. Arrays can be single or multidimensional for organizing complex data. We used 2d arrays to represent matrices and multiply them.

A1.

```
#include<stdio.h>
#include<conio.h>

void main(){
    int m = 5;
    int arr1[m];
    for (int i = 0; i<m;i++){
        printf("Please enter the number: ");
        scanf("%d", &arr1[i]);
    }
    for (int i = 0;i<m-1;i++){
        for(int j = 0;j<m-1;j++){
            if (arr1[j] < arr1[j+1]){
                int temp = arr1[j];
                arr1[j] = arr1[j+1];
                arr1[j+1] = temp;
            }
        }
    }
    printf("Descending Order\n");
    for (int i = 0;i<m;i++){
        printf("%d\t", arr1[i]);
    }
}
```

```
"C:\Users\tarun\Downloads\T" + ▾
Please enter the number: 5
Please enter the number: 3
Please enter the number: 1
Please enter the number: 7
Please enter the number: 4
Descending Order
7      5      4      3      1
Process returned 5 (0x5)   execution time : 5.587 s
Press any key to continue.
|
```

A2.

```
#include<stdio.h>
#include<conio.h>

void main(){
    int m = 3;
    int n = m;
    int arr1[m][n];
    printf("Please enter array 1\n");
    for (int i = 0; i<m;i++){
        for (int j = 0; j<n;j++){
            int a = 0;
            printf("Please enter the number: ");
            scanf("%d", &a);
            arr1[i][j] = a;
        }
    }
    int arr2[m][n];
    printf("Please enter array 2\n");
    for (int i = 0; i<m;i++){
        for (int j = 0; j<n;j++){
            int a = 0;
            printf("Please enter the number: ");
            scanf("%d", &a);
            arr2[i][j] = a;
        }
    }
}
```

```

int ans[m][n];
for (int i = 0; i<m;i++){
    for (int j = 0; j<n;j++){
        int sum = 0;
        for (int z = 0; z<n; z++){
            sum += arr1[z][j]*arr2[j][z];
        }
        ans[i][j] = sum;
    }
}
printf("Product of array 1 and 2\n");
for (int i = 0; i<m;i++){
    for (int j = 0; j<n;j++){
        printf("%d\t", ans[i][j]);
    }
    printf("\n");
}
}

```

```

"C:\Users\tarun\Downloads\T" + 
Please enter array 1
Please enter the number: 1
Please enter the number: 2
Please enter the number: 3
Please enter the number: 1
Please enter the number: 2
Please enter the number: 3
Please enter the number: 1
Please enter the number: 2
Please enter the number: 3
Please enter array 2
Please enter the number: 1
Please enter the number: 2
Please enter the number: 3
Please enter the number: 1
Please enter the number: 2
Please enter the number: 3
Please enter the number: 1
Please enter the number: 2
Please enter the number: 3
Product of array 1 and 2
6      12      18
6      12      18
6      12      18
Process returned 3 (0x3)   execution time : 15.114 s
Press any key to continue.

```

Conclusion

In conclusion, I learned that 1D and 2D arrays in C are powerful tools for storing and organizing data. A 1D array helps manage linear data efficiently, while a 2D array arranges data in rows and columns for easier processing and better program structure.