Experiment No-08: Two dimensional arrays.

Objectives

- Write C programs using 2D arrays.
- Solve various matrix problems.

Example 1: A C++ program to multiply two matrices.

```
#include<iostream>
using namespace std;

int main(){

int a[10][10],b[10][10],result[10][10],multiplied[10][10],r,c,i,j,k;

cout<<"Enter the number of rows = ";
cin>>r;
cout<<"Enter the number of columns = ";
cin>>c;
//first matrix
cout<<"Enter the first matrix elements"<<endl;
for(i=0;i<r;i++)
{
    for(j=0;j<c;j++)
    {
        cin>>a[i][j];
    }
}
//second matrix
cout<<"Enter the second matrix elements"<<endl;
for(i=0;i<r;i++)
{
    for(j=0;j<c;j++)
    {
    cin>>b[i][j];
    }
}
// matrix multiplication
cout<<"Multiply of the matrix ="<<endl;
for(i=0;i<r;i++)
{
    for(j=0;j<c;j++)
    {
        result[i][j]=0;
        for(k=0;k<c;k++)
        {
            result[i][j]+=a[i][k]*b[k][j];
        }
```

```
            multiplied[i][j] = result[i][j];
        }
    }
    //for printing result
    for(i=0;i<r;i++)
    {
        for(j=0;j<c;j++)
        {
            cout<<multiplied[i][j]<<" ";
        }
        cout<<endl;
    }

    return 0;
    }
```

Example 2: A C++ Program to find the upper triangular matrix.

Upper Triangular Matrix: A special square matrix whose all elements below the main diagonal are zero.

```
#include<iostream>
using namespace std;

int main()
{
    int arr[10][10];
    int row, col, total_row, total_col, isupper;

    // Input elements in matrix from user
    cout<<"Enter the rows and columns: ";
    cin>>total_row>>total_col;
    for(row=0; row<total_row; row++)
    {
        for(col=0; col<total_col; col++)
        {
            cin>>arr[row][col];
        }
    }

    // Check Upper triangular matrix condition
    isupper = 1;
    for(row=0; row<total_row; row++)
    {
        for(col=0; col<total_col; col++)
        {

            //If elements below the main diagonal (col<row)
            //is not equal to zero then it is not upper triangular
                matrix
```

```
            if(col<row && arr[row][col]!=0)
            {
                isupper = 0;
            }
        }
    }


    if(isupper == 1)
    {
        cout<<"\nThe matrix is Upper triangular matrix."<<endl;

    }
    else
    {
        cout<<"\nThe matrix is not Upper triangular matrix."<<endl;
    }

    return 0;
}
```

Practice Exercise

1. Write a C++ program to add two matrices.
2. Write a C++ program to subtract two matrices.
3. Write a C++ program to perform Scalar matrix multiplication.
4. Write a C++ program to check whether two matrices are equal or not.
5. Write a C++ program to find the sum of the main diagonal elements of a matrix.
6. Write a C++ program to find the sum of the minor diagonal elements of a matrix.
7. Write a C++ program to find the lower triangular matrix.
8. Write a C++ program to find the sum of the upper triangular matrix.
9. Write a C++ program to check whether a matrix is sparse or not.
10. Write a C++ program to check whether a matrix is an identity matrix or not.