NAME: SMRITI

ROLL NO: 102116094

CLASS: 2CS11

QUESTION-1: (a)

```
# include<iostream>
# define max 4
using namespace std;
int main()
{
    int i,j,a[max];
    cout<<"enter the elements row major"<<endl;
    for(i=0; i<max ;i++)
    {
        cin>>a[i];
    }
    cout<<"the matrix is"<<endl;
    for(int i=0;i<max;i++)
    {
        for(int j=0;j<max;j++)
        {
            if(i==j)
                cout<<a[i];
            else
                cout<<"0";
        }
        cout<<endl;
    }
return 0;
}
```

OUTPUT:

```
PS C:\Users\Smriti\Desktop\a1> cd "c:\Users\Smriti\Desktop\a1\" ;
enter the elements row major
1 5 7 9
the matrix is
1 0 0 0
0 5 0 0
0 0 7 0
0 0 0 9
PS C:\Users\Smriti\Desktop\a1>
```

QUESTION-1: (b)

```cpp
#include<iostream>
# define max 4
using namespace std;
int main()
{
    int i,j,k=0,size=3*max-2,a[size];
    cout<<"enter the elements (row major)"<<endl;
    for (int i=0;i<size;i++)
    {
        cin>>a[i];
    }
    cout<<" the matrix is"<<endl;
    for (int i=0; i<max;i++)
    {
        for (int j=0;j<max;j++)
        {
            if( i-j==1 || i-j==0  || i-j==-1)
            {
                cout<<a[k]<<" ";
                k++;
            }
            else
            cout<<"0 ";
        }
        cout<<endl;
    }
    return 0;
}
```

OUTPUT:

```
enter the elements (row major)
1 2 3 4 5 6 7 8 9
10
 the matrix is
1 2 0 0
3 4 5 0
0 6 7 8
0 0 9 10
PS C:\Users\Smriti\Desktop\a1>
```

QUESTION-1(C & D):

```cpp
#include<iostream>
# define max 4
using namespace std;
int main()
{
    int i,j,k=0,size=(max*(max+1))/2,a[size];
    cout<<"enter the elements (row major)"<<endl;
    for(int i=0;i<size;i++)
    {
        cin>>a[i];
    }
    cout<<"the upper triangle matrix is"<<endl;
    for(int i=0;i<max;i++)
    {
        for(int j=0;j<max;j++)
        {
            if(i<=j)
            {
                cout<<a[k]<<" ";
                k++;
            }
            else
                cout<<"0 ";
        }
        cout<<endl;
    }
    k=0;
    cout<<"the lower triangle matrix is"<<endl;
    for(int i=0;i<max;i++)
    {
        for(int j=0;j<max;j++)
        {
            if(i>=j)
            {
                cout<<a[k]<<" ";
                k++;
            }
            else
            cout<<"0 ";
        }
        cout<<endl;
    }
    return 0;
}
```

OUTPUT:

```
PS C:\Users\Smriti\Desktop\a1> cd "c:\Users\Smriti\Desktop\a1\" ;
enter the elements (row major)
1 2 3 4 5 6 7 8
9 10
the upper triangle matrix is
1 2 3 4
0 5 6 7
0 0 8 9
0 0 0 10
the lower triangle matrix is
1 0 0 0
2 3 0 0
4 5 6 0
7 8 9 10
PS C:\Users\Smriti\Desktop\a1>
```

QUESTION-1 (e)

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

void symmetric(int array[], int n)
{
    int k=0;
    int matrix[n][n];
    for(int i =0;i<n;i++)
    {
        for(int j =0;j<n;j++)
        {
            if(i==j)
            {
                matrix[i][j] = array[k];
                k++;
            }
            else if(i<j)
            {
                matrix[i][j] = array[k];
                matrix[j][i] = array[k];
                k++;
            }
        }
    }

    for(int i =0;i<n;i++)
    {
```

```
        for(int j=0;j<n;j++)
        {
            cout << matrix[i][j] << " ";
        }
        cout << endl;
    }

}

int main()
{
    int n = 3;
    int arr[6] = {1,2,3,4,5,6};
    symmetric(arr, n);

    return 0;
}
```

OUTPUT:

```
PS C:\Users\Smriti\Desktop\a1> cd "c:\
5 4 3
4 2 1
3 1 0
PS C:\Users\Smriti\Desktop\a1>
```

QUESTION:2->(a)

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

void sum(int n)
{

    int matrix[n][n];
    for (int i = 0; i < n; i++)
    {
        for (int j = 0; j < n; j++)
        {
            cin >> matrix[i][j];
        }
    }
```

```
    for (int i = 0; i < n; i++)
    {
        for (int j = 0; j < n; j++)
        {
            cout << matrix[i][j] << " ";
        }
        cout << endl;
    }

    for (int i = 0; i < n; i++)
    {
        int sum1 = 0;
        for (int j = 0; j < n; j++)
        {
            sum1 = sum1 + matrix[i][j];
        }
        cout << i << "th row sum = " << sum1 << endl;
    }
    for (int i = 0; i < n; i++)
    {
        int sum2 = 0;
        for (int j = 0; j < n; j++)
        {
            sum2 = sum2 + matrix[j][i];
        }
        cout << i << "th column sum = " << sum2 << endl;
    }
}

int matrix[10][10];

void sparsematrix(int n, int m)
{
    for (int i = 0; i < n; i++)
    {
        for (int j = 0; j < m; j++)
        {
            cin >> matrix[i][j];
        }
        cout<<endl;
    }

    for (int i = 0; i < n; i++)
    {
        for (int j = 0; j < m; j++)
        {
            cout << matrix[i][j] << " ";
        }
```

```
        cout << endl;
    }
}

void compactMatrix(int n, int m)
{
    int size = 0;
    for (int i = 0; i < n; i++)
    {
        for (int j = 0; j < m; j++)
        {
            if (matrix[i][j] != 0)
            {
                size++;
            }
        }
    }

    int compactMatrix[size][3];

    int k = 0;

    for (int i = 0; i < n; i++)
    {
        for (int j = 0; j < m; j++)
        {
            if (matrix[i][j] != 0)
            {
                compactMatrix[k][0] = i;
                compactMatrix[k][1] = j;
                compactMatrix[k][2] = matrix[i][j];
                k++;
            }
        }
    }

    for (int i = 0; i < size; i++)
    {
        for (int j = 0; j < 3; j++)
        {
            cout << compactMatrix[i][j] << " ";
        }
        cout << endl;
    }
}

void transpose(int n, int m)
{
```

```
    for (int i = 0; i < n; i++)
    {
        for (int j = 0; j < m; j++)
        {
            if (i < j)
            {

                int temp;
                temp = matrix[i][j];
                matrix[i][j] = matrix[j][i];
                matrix[j][i] = temp;
            }
        }
    }

    for (int i = 0; i < m; i++)
    {
        for (int j = 0; j < n; j++)
        {
            cout << matrix[i][j] << " ";
        }
        cout << endl;
    }
}

int main()
{
    int n, m;
    cout << "Enter the no. of rows = ";
    cin >> n;
    cout << "Enter the no. of columns = ";
    cin >> m;

    cout << "Enter the elements:" << endl;


    sparsematrix(n, m);

    cout << endl;
    cout << endl;

    compactMatrix(n, m);

    cout << endl;
    cout << endl;

    transpose(n, m);
```

```
    cout << endl;
    cout << endl;

    compactMatrix(m, n);

    return 0;
}
```

```
Enter the no. of rows = 3
Enter the no. of columns = 3
Enter the elements:
 1 2 3 4 5 6 7 8 9



1 2 3
4 5 6
7 8 9


0 0 1
0 1 2
0 2 3
1 0 4
1 1 5
1 2 6
2 0 7
2 1 8
2 2 9


1 4 7
2 5 8
3 6 9


0 0 1
0 1 4
0 2 7
1 0 2
```

```
0 0 1
0 1 4
0 2 7
1 0 2
1 1 5
1 2 8
2 0 3
2 1 6
2 2 9
PS C:\Users\Smriti\Desktop\a1>
```

QUESTION-2 (B)

```
#include <iostream>
#include <stdlib.h>
using namespace std;
void set_value(int *arr, int n, int row, int col)
{
    int r1, c1, val;
    for (int i = 0; i < n; i++)
    {
        cout << "Enter the row column and value of element " << i + 1 << ": ";
        cin >> r1 >> c1 >> val;
        if (r1 > row)
        {
            cout << "Entered row number is not valid....\n";
            cout << "Enter again........\n";
            i--;
            continue;
        }
        else if (c1 > col)
        {
            cout << "Entered column number is not valid....\n";
            cout << "Enter again........\n";
            i--;
            continue;
        }
```

```
        *(arr + i) = r1;
        *(arr + i + n) = c1;
        *(arr + i + 2 * n) = val;
    }
}
int binarySearch(int *arr, int size, int element)
{
    int temp;
    for (int i = 0; i < size - 1; i++)
    {
        if (*(arr + i) > *(arr + i + 1))
        {
            for (int j = 0; j < size - 1; j++)
            {
                if (*(arr + j) > *(arr + i + 1))
                {
                    temp = *(arr + j);
                    *(arr + j) = *(arr + i + 1);
                    *(arr + i + 1) = temp;
                }
            }
        }
    }
    int low = 0, high = size - 1, mid = (low + high) / 2;
    while (high >= low)
    {
        if (*(arr + low) == element)
        {
            return true;
        }
        else if (*(arr + high) == element)
        {
            return true;
        }
        else if (*(arr + mid) == element)
        {
            return true;
        }
        else if (low == mid)
        {
            return false;
        }
        else if (element > *(arr + mid))
        {
            low = mid;
            mid = (low + high) / 2;
        }
        else if (element < *(arr + mid))
```

```
        {
            high = mid;
            mid = (low + high) / 2;
        }
    }
}
void add_matrix(int *arr1, int *arr2, int n1, int n2)
{
    int *sumArr = (int *)(malloc(3 * (n1 + n2) * sizeof(int)));
    int *temp, *note = (int *)(malloc(n1 * sizeof(int)));
    int index = 0, check, sumIndex = 0;
    for (int i = 0; i < n1; i++)
    {
        check = 0;
        for (int j = 0; j < n2; j++)
        {
            if (*(arr1 + i) == *(arr2 + j))
            {
                if (*(arr1 + i + n1) == *(arr2 + j + n2))
                {
                    *(sumArr + i) = *(arr1 + i);
                    *(sumArr + i + (n1 + n2)) = *(arr1 + i + n1);
                    *(sumArr + i + 2 * (n1 + n2)) = *(arr1 + i + 2 * n1) +
*(arr2 + j + 2 * n2);
                    *(note + index) = j;
                    index++;
                    check++;
                    sumIndex++;
                }
            }
        }
        if (check == 0)
        {
            *(sumArr + i) = *(arr1 + i);
            *(sumArr + i + (n1 + n2)) = *(arr1 + i + n1);
            *(sumArr + i + 2 * (n1 + n2)) = *(arr1 + i + 2 * n1);
            sumIndex++;
        }
    }
    check = 0;
    for (int i = 0; i < n2; i++)
    {
        if (binarySearch(note, index, i))
        {
            check++;
            continue;
        }
        else
```

```
        {
            *(sumArr + n1 + i - check) = *(arr2 + i);
            *(sumArr + n1 + i - check + (n1 + n2)) = *(arr2 + i + n2);
            *(sumArr + n1 + i - check + 2 * (n1 + n2)) = *(arr2 + i + 2 * n2);
            sumIndex++;
        }
    }
    cout << "Printing the matrix.....\n";
    cout << "Row\tColumns\tValue\n";
    for (int i = 0; i < sumIndex; i++)
    {
        cout << *(sumArr + i) << "\t" << *(sumArr + i + n1 + n2) << "\t" <<
*(sumArr + i + 2 * (n1 + n2));
        cout << endl;
    }
}
int main()
{
    int row, col, n1, n2;

    cout << "Enter the number of rows and columns for both the matrix: ";
    cin >> row >> col;

    cout << "Enter the number of non-zeros elements in matrix 1: ";
    cin >> n1;

    int *arr1 = (int *)(malloc(3 * n1 * sizeof(int)));
    set_value(arr1, n1, row, col);

    cout << "Enter the number of non-zeros elements in matrix 2: ";
    cin >> n2;

    int *arr2 = (int *)(malloc(3 * n2 * sizeof(int)));
    set_value(arr2, n2, row, col);
    add_matrix(arr1, arr2, n1, n2);

    return 0;
}
```

QUESTION:3

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

int main()
{
    int n , m;
    cout<<"Enter the no.of rows = ";
    cin>>n;
    cout<<"Enter the no. of columns = ";
    cin>>m;
    int arr[n][m];

    cout<<endl;
    cout<<"Enter the elemnts of the matrix:"<<endl;
    for(int i=0; i<n; i++)
    {
        for(int j=0; j<m; j++)
        {
            cin>>arr[i][j];
        }
    }

    cout<<endl;
    cout<<"The matrix is:"<<endl;
    for(int i=0; i<n; i++)
    {
        for(int j=0; j<m; j++)
        {
            cout<<arr[i][j]<<" ";
        }
        cout<<endl;
    }
    int sum=0;

    for(int i=0; i<n; i++)
    {
        for(int j=0; j<m; j++)
        {
            sum = sum + arr[i][j];
        }
         cout<<"Sum of row "<< i <<" = "<<sum<<endl;
         sum=0;
    }
    cout<<endl;
```

```
     for(int i=0; i<n; i++)
    {
        for(int j=0; j<m; j++)
        {
            sum = sum + arr[j][i];
        }
         cout<<"Sum of column "<< i <<" = "<<sum<<endl;
         sum=0;
    }
    return 0;
}
```

OUTPUT:

```
PS C:\Users\Smriti\Desktop\a1> cd "c:\Users\Smriti\Desktop\a1\" ;
Enter the no.of rows = 3
Enter the no. of columns = 3

Enter the elemnts of the matrix:
9 8 7 6 5 4 3 2 1

The matrix is:
9 8 7
6 5 4
3 2 1
Sum of row 0 = 24
Sum of row 1 = 15
Sum of row 2 = 6

Sum of column 0 = 18
Sum of column 1 = 15
Sum of column 2 = 12
PS C:\Users\Smriti\Desktop\a1>
```

QUESTION:4

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

int main()
{
    int n,m;
    cout<<"Enter the no. of rows = ";
    cin>>n;
    cout<<"Enter the no. of columns = ";
    cin>>m;
```

```
int arr[n][m];
int rowmin[n];
int colmax[m];
cout<<endl;
cout<<"Enter the elements of the array:"<<endl;

for(int i=0; i<n; i++)
{
    for(int j=0; j<m; j++)
    {
        cin>>arr[i][j];
    }
}

cout<<endl;
cout<<"The matrix is:"<<endl;
 for(int i=0; i<n; i++)
 {
    for(int j=0; j<m; j++)
    {
       cout<<arr[i][j]<<" ";
    }
    cout<<endl;
 }

 cout<<endl;

 int temp , b=0;

  for (int i = 0; i<n; i++)
{
    temp = arr[i][0];
    for (int j = 1; j<m; j++)
    {
        if (temp > arr[i][j])
        {
            temp = arr[i][j];
        }
    }
    rowmin[b] = temp;
    b++;
}
b = 0;
for (int i = 0; i<m; i++)
{
    temp = arr[0][i];
    for (int j = 1; j<n; j++)
```

```
        {
            if (temp < arr[j][i])
            {
                temp = arr[j][i];
            }
        }
        colmax[b] = temp;
        b++;
    }
    b = 0;
    cout<<endl;
    for (int i = 0; i<n; i++)
    {
        for (int j = 0; j<m; j++)
        {
            if (rowmin[i] == colmax[j])
            {
                cout << "The saddle point is: " << rowmin[i] << endl;
                b++;
                break;
            }
        }
    }
    if (b == 0)
    {
        cout << "No Saddle point exists\n";
    }
    return 0;
}
```

OUTPUT:

```
Enter the no. of rows = 3
Enter the no. of columns = 3

Enter the elements of the array:
1 3 5 7 9 8 4 2 0cd "c:\Users\Smriti\Desktop\a1\" ;
} ; if ($?) { .\tempCodeRunnerFile }

The matrix is:
1 3 5
7 9 8
4 2 0


The saddle point is: 7
PS C:\Users\Smriti\Desktop\a1>
```

QUESTION:5->

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

int main()
{
    int m,n;
    cout<<"Enter the no. of rows = ";
    cin>>m;
    cout<<"Enter the no. of columns = ";
    cin>>n;

    int arr[m][n];
    cout<<endl;
    cout<<"Enter the elements of the array:"<<endl;

    for(int i=0; i<m; i++)
    {
        for(int j=0; j<n; j++)
        {
            cin>>arr[i][j];
        }
    }

    cout<<endl;
    cout<<"The matrix is:"<<endl;
     for(int i=0; i<m; i++)
     {
        for(int j=0; j<n; j++)
        {
           cout<<arr[i][j]<<" ";
        }
        cout<<endl;
     }

     cout<<endl;


    int i, k = 0, l = 0;

    cout<<"The Spiral Matrix is:"<<endl;

  while (k < m && l < n) {

        for (i = l; i < n; ++i) {
            cout << arr[k][i] << " ";
```

```
        }
        k++;


        for (i = k; i < m; ++i) {
            cout << arr[i][n - 1] << " ";
        }
        n--;

        if (k < m) {
            for (i = n - 1; i >= l; --i) {
                cout << arr[m - 1][i] << " ";
            }
            m--;
        }


        if (l < n) {
            for (i = m - 1; i >= k; --i) {
                cout << arr[i][l] << " ";
            }
            l++;
        }
    }
    return 0;
}
```

OUTPUT:

```
PS C:\Users\Smriti\Desktop\a1> cd "c:\Users\Smriti\Desktop\a1\" ;
Enter the no. of rows = 3
Enter the no. of columns = 3

Enter the elements of the array:
1 4 7 9 4 6 0 5 3

The matrix is:
1 4 7
9 4 6
0 5 3

The Spiral Matrix is:
1 4 7 6 3 5 0 9 4
PS C:\Users\Smriti\Desktop\a1>
```

QUESTION:6->

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

void spiral(int n)
{
    int row = n;
    int col = n;
    int size = n*n;
    int arr[size];
    for(int i = 0;i<size;i++)
    {
        arr[i] = i+1;
    }

    int matrix[n][n];
    int top =0;
    int bottom = row-1;
    int right = col-1;
    int left =0;

    int dir =1;
    int k=0;
    while(top<=bottom && left <= right)
    {

        if(dir == 1)
        {
            for(int i = left ; i<=right; i++)
            {
                matrix[top][i] = arr[k];
                k++;
            }
             top++;
             dir = 2;
        }

        else if(dir == 2)
        {
            for(int i = top ; i<=bottom;i++)
            {
                matrix[i][right] = arr[k];
                k++;
            }
            right--;
            dir = 3;
```

```
        }

        else if(dir == 3)
        {
            for(int i = right; i>= left; i--)
            {
                matrix[bottom][i] = arr[k];
                k++;
            }
            bottom--;
            dir = 4;
        }

        else if(dir == 4)
        {
            for(int i = bottom; i>=top; i--)
            {
                matrix[i][left] = arr[k];
                k++;
            }
            left++;
            dir = 1;
        }
    }

    for(int i = 0;i<n;i++)
    {
        for(int j =0;j<n ;j++)
        {
            cout << matrix[i][j] << " ";
        }
        cout << endl;
    }

}

int main()
{
  int n;
  cout<<"Enter the integer = ";
  cin>>n;
    spiral(n);
    return 0;
}
```

OUTPUT:

```
PS C:\Users\Smriti\Desktop\a1> cd "c:\Users\Smriti\Desktop\a1\" ;
Enter the integer = 4
1 2 3 4
12 13 14 5
11 16 15 6
10 9 8 7
PS C:\Users\Smriti\Desktop\a1>
```