**1(a)**

#include<iostream>

#define MAX 4

using namespace std;

int main()

{

int i,j,a[MAX];

int k=0;

cout<<"enter elements in diagonal"<<endl;

for(i=0 ; i<MAX ; i++)

{

cin>>a[i];

}

cout<<"matrix is :"<<endl;

for(i=0; i<MAX; i++)

{

for(j=0; j<MAX; j++)

{

if(i==j)

{

cout<<a[k]<<"\t";

k++;

}

else

{

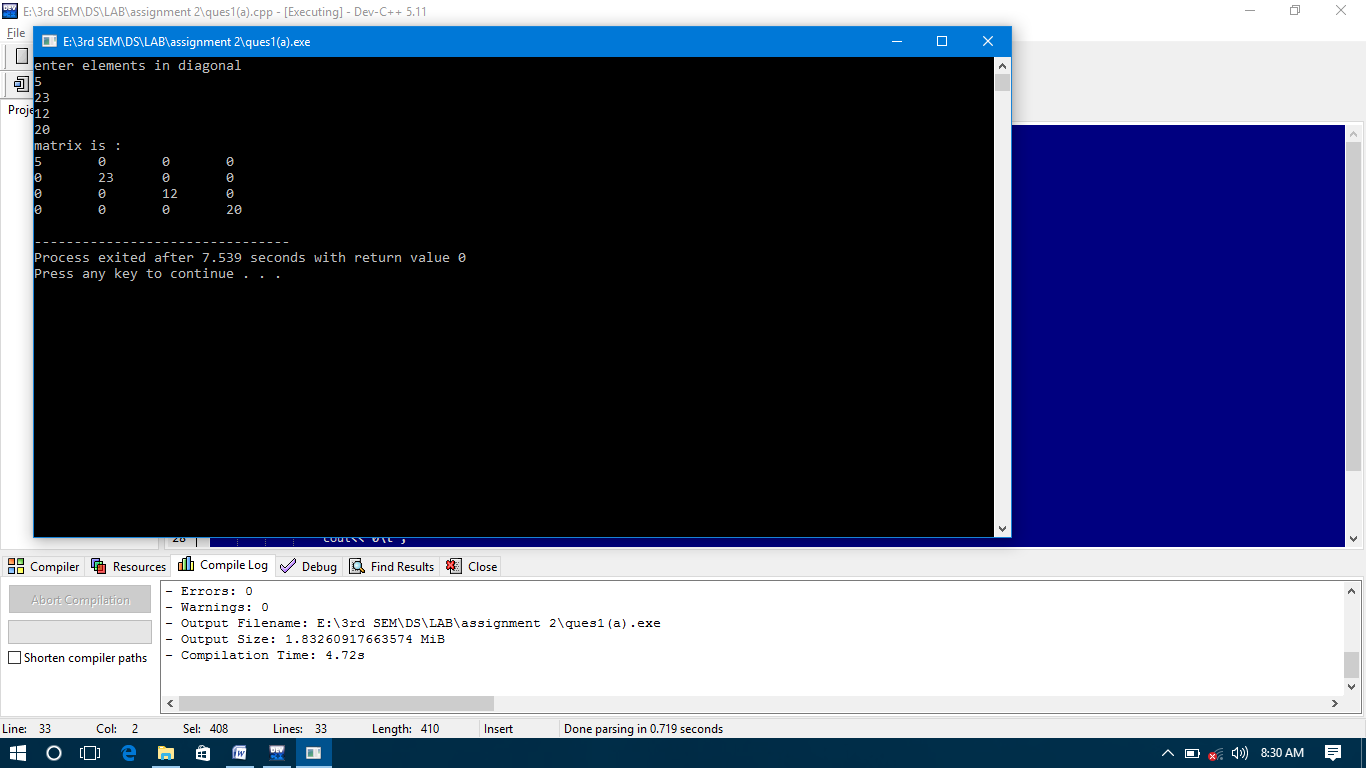
cout<<"0\t";

}

}

cout<<endl;

}}



**1(b)**

#include<iostream>

#define MAX 4

using namespace std;

int main()

{

int i,j,size;

size=3\*MAX-2;

int a[size];

int k=0;

cout<<"enter elements for tridiagonal matrix{row major}"<<endl;

for(i=0 ; i<size ; i++)

{

cin>>a[i];

}

cout<<"matrix is :"<<endl;

for(i=0; i<MAX; i++)

{

for(j=0; j<MAX; j++)

{

if(i-j==0 || i-j==1 || i-j==-1)

{

cout<<a[k]<<"\t";

k++;

}

else

{

cout<<"0\t";

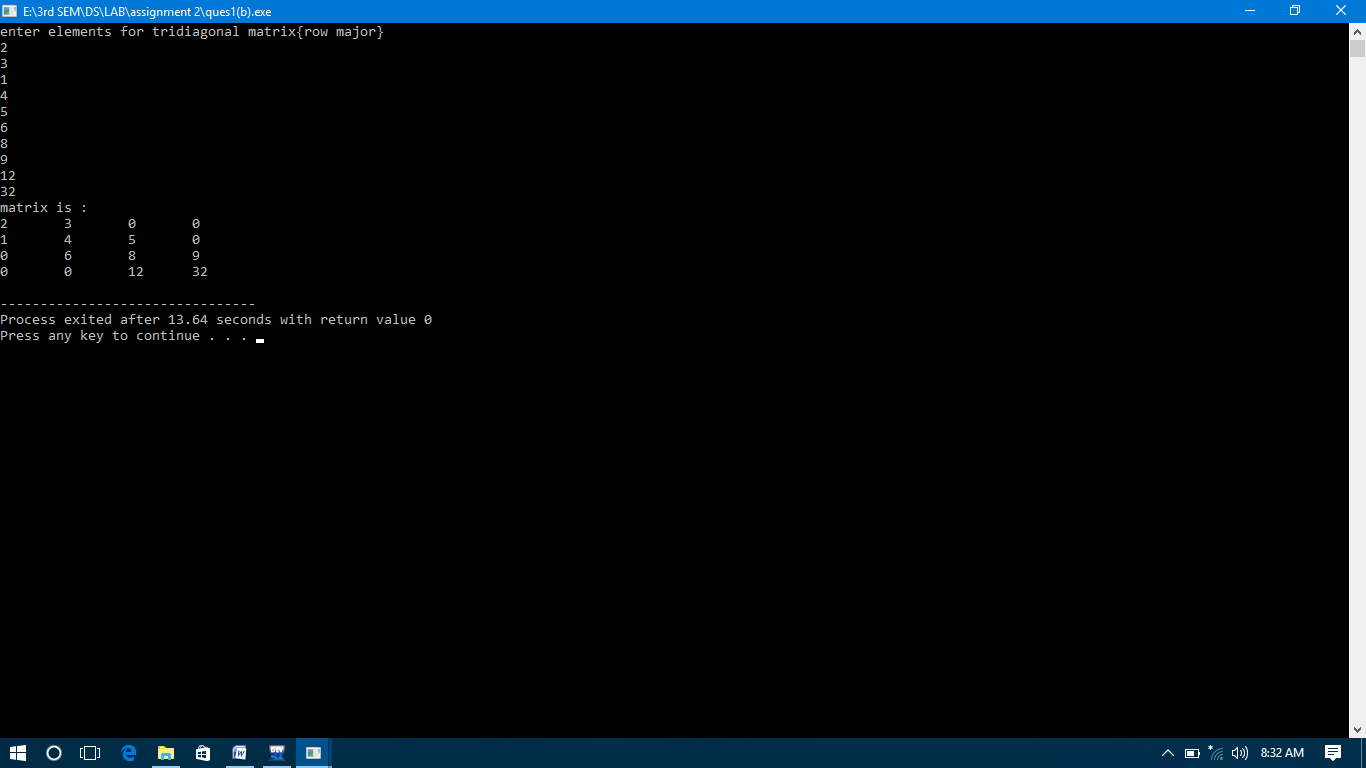
}

}

cout<<endl;

}

}



**1(c)**

#include<iostream>

#define MAX 4

using namespace std;

int main()

{

int i,j,size;

size=3\*MAX-2;

int a[size];

int k=0;

cout<<"enter elements for upper triangular matrix{row major}"<<endl;

for(i=0 ; i<size ; i++)

{

cin>>a[i];

}

cout<<"matrix is :"<<endl;

for(i=0; i<MAX; i++)

{

for(j=0; j<MAX; j++)

{

if(i<j || i==j)

{

cout<<a[k]<<"\t";

k++;

}

else

{

cout<<"0\t";

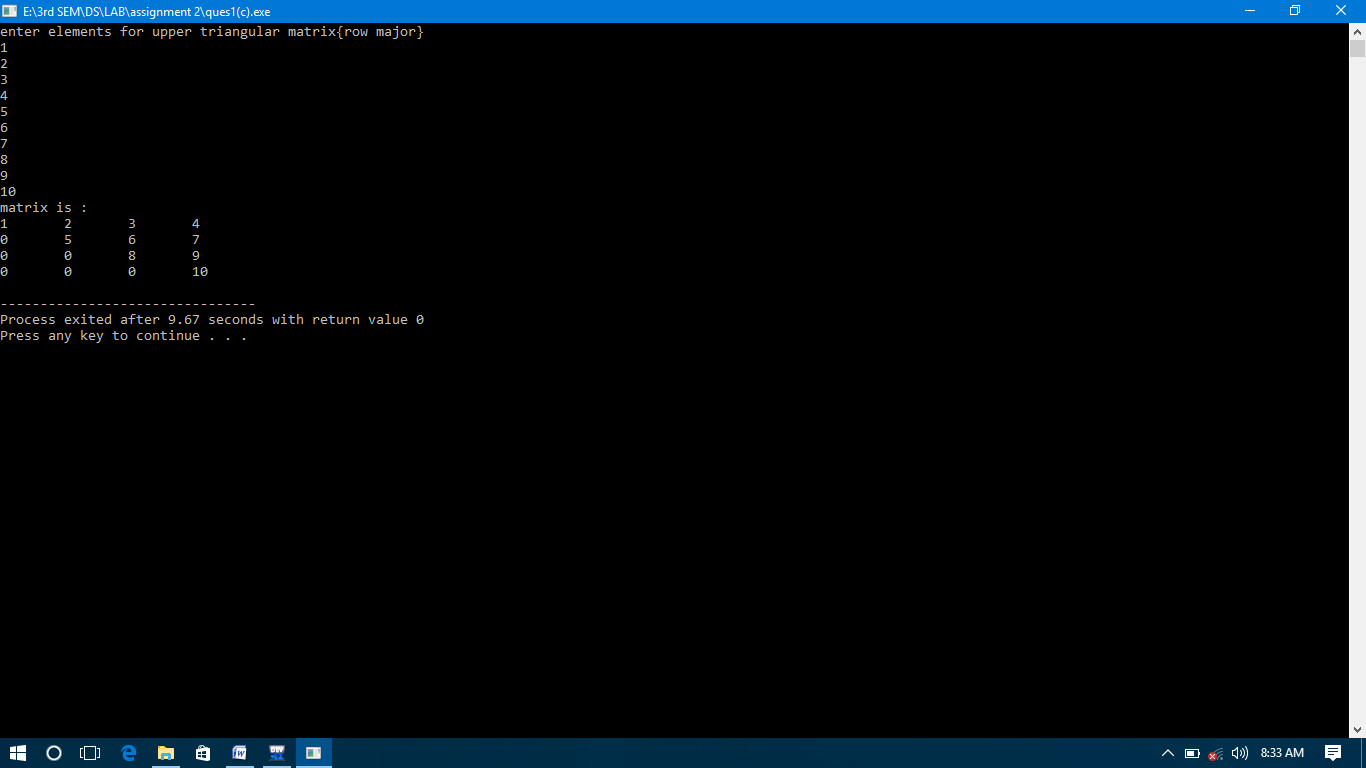
}

}

cout<<endl;

}

}



**1(d)**

#include<iostream>

#define MAX 4

using namespace std;

int main()

{

int i,j,size;

size=3\*MAX-2;

int a[size];

int k=0;

cout<<"enter elements for lower triangular matrix{row major}"<<endl;

for(i=0 ; i<size ; i++)

{

cin>>a[i];

}

cout<<"matrix is :"<<endl;

for(i=0; i<MAX; i++)

{

for(j=0; j<MAX; j++)

{

if(i>j || i==j)

{

cout<<a[k]<<"\t";

k++;

}

else

{

cout<<"0\t";

}

}

cout<<endl;

}

}

**1(e)**

#include <iostream>

using namespace std;

int main(){

int n;

cout<<"enter no. of rows and column in symmetric matrix\n";

cin>>n;

int b=n\*(n+1)/2;

int a[b];

cout<<"enter elements row major for lower triangular part";

for(int i=0;i<b;i++)

{

cout<<"enter element\n";

cin>>a[i];

}

int k=0;

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

{ for(int j=0;j<n;j++)

{

if(i>=j){

cout<<a[((i\*(i+1))/2)+j]<<" ";

}

else

cout<<a[((j\*(j+1))/2)+i]<<" ";

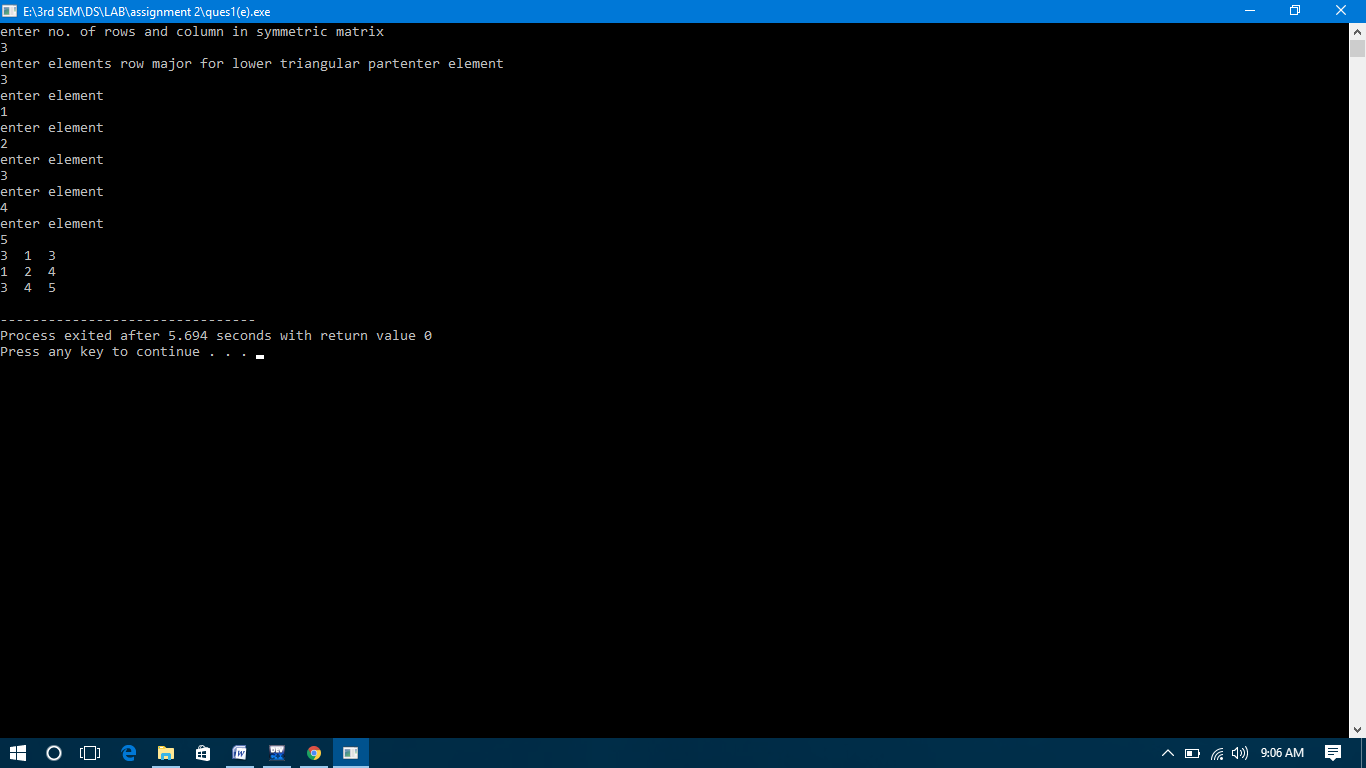
}

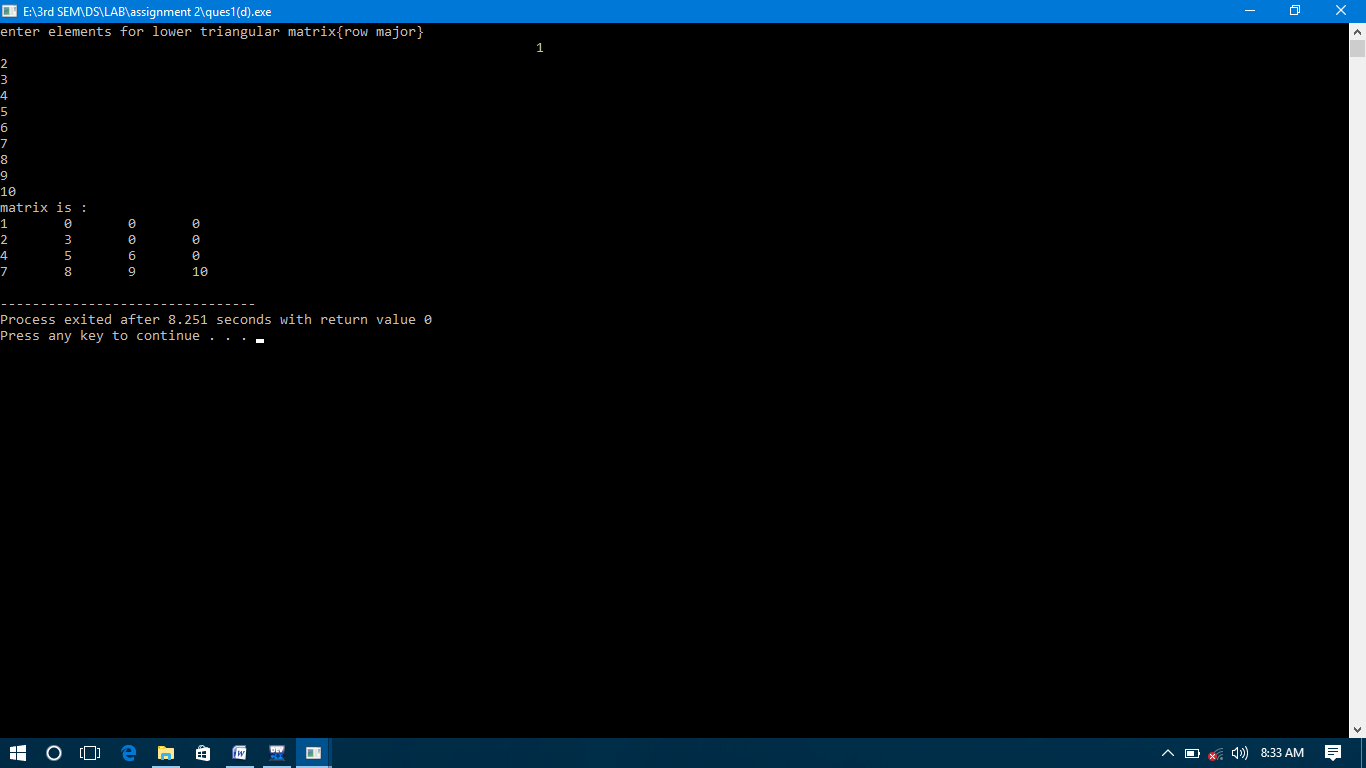
cout<<"\n";

}

return 0;

}





**Q2**

#include<iostream>

#define max 12

using namespace std;

void readsp(int b[][3])

{

int i;

cout<<"enter number of rows and columns:"<<endl;

cin>>b[0][0]>>b[0][1];

cout<<"enter number of values"<<endl;

cin>>b[0][2];

cout<<"enter triplet (row,column,value) : "<<endl;

for(i=1; i<=b[0][2]; i++)

{

cin>>b[i][0]>>b[i][1]>>b[i][2];

}

}

void printsp(int b[][3])

{

int i,j,k=1;

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

for(i=0; i<b[0][0]; i++)

{

for(j=0; j<b[0][1]; j++)

{

if( i==b[k][0] && j==b[k][1] )

{

cout<<b[k][2]<<"\t";

k++;

}

else

{

cout<<"0\t";

}

}

cout<<endl;

}

}

void transposesp(int a[][3], int b[][3])

{

int i,j,k=1;

b[0][0] = a[0][1];

b[0][1] = a[0][0];

b[0][2] = a[0][2];

if(b[0][2] > 0)

{ for(i=0;i<a[0][1];i++)

{ for(j=1;j<=a[0][2];j++)

{ if(a[j][1]==i)

{ b[k][0] = a[j][1];

b[k][1] = a[j][0];

b[k][2] = a[j][2];

k++;

}

}

}

}

}

int addsp(int sp1[][3], int sp2[][3], int sp3[][3])

{ int k1, k2, k3, tot1, tot2;

if(sp1[0][0] == sp2[0][0] && sp1[0][1] == sp2[0][1])

{ tot1 = sp1[0][2]; tot2 = sp2[0][2];

k1 = k2 = k3 = 1;

while ( k1 <= tot1 && k2 <= tot2)

{ if ( sp1[k1][0] < sp2[k2][0] )

{ sp3[k3][0] = sp1[k1][0];

sp3[k3][1] = sp1[k1][1];

sp3[k3][2] = sp1[k1][2];

k3++; k1++;

}

else if ( sp1[k1][0] > sp2[k2][0] )

{ sp3[k3][0] = sp2[k2][0];

sp3[k3][1] = sp2[k2][1];

sp3[k3][2] = sp2[k2][2];

k3++; k2++;

}

else

{ if ( sp1[k1][1] < sp2[k2][1] )

{ sp3[k3][0] = sp1[k1][0];

sp3[k3][1] = sp1[k1][1];

sp3[k3][2] = sp1[k1][2];

k3++; k1++;

}

else if ( sp1[k1][1] > sp2[k2][1] )

{ sp3[k3][0] = sp2[k2][0];

sp3[k3][1] = sp2[k2][1];

sp3[k3][2] = sp2[k2][2];

k3++; k2++;

}

else

{ sp3[k3][0] = sp2[k2][0];

sp3[k3][1] = sp2[k2][1];

sp3[k3][2] = sp1[k1][2] + sp2[k2][2];

k3++; k2++; k1++;

}

}

}

while ( k1 <=tot1 )

{ sp3[k3][0] = sp1[k1][0];

sp3[k3][1] = sp1[k1][1];

sp3[k3][2] = sp1[k1][2];

k3++; k1++;

}

while ( k2 <= tot2 )

{ sp3[k3][0] = sp2[k2][0];

sp3[k3][1] = sp2[k2][1];

sp3[k3][2] = sp2[k2][2];

k3++; k2++;

}

sp3[0][0] = sp1[0][0];

sp3[0][1] = sp1[0][1];

sp3[0][2] = k3-1;

}

else

printf("\nInvalid Dimensions.\n");

}

int main()

{

int a[max][3],b[max][3],c[max][3];

readsp(a);

printsp(a);

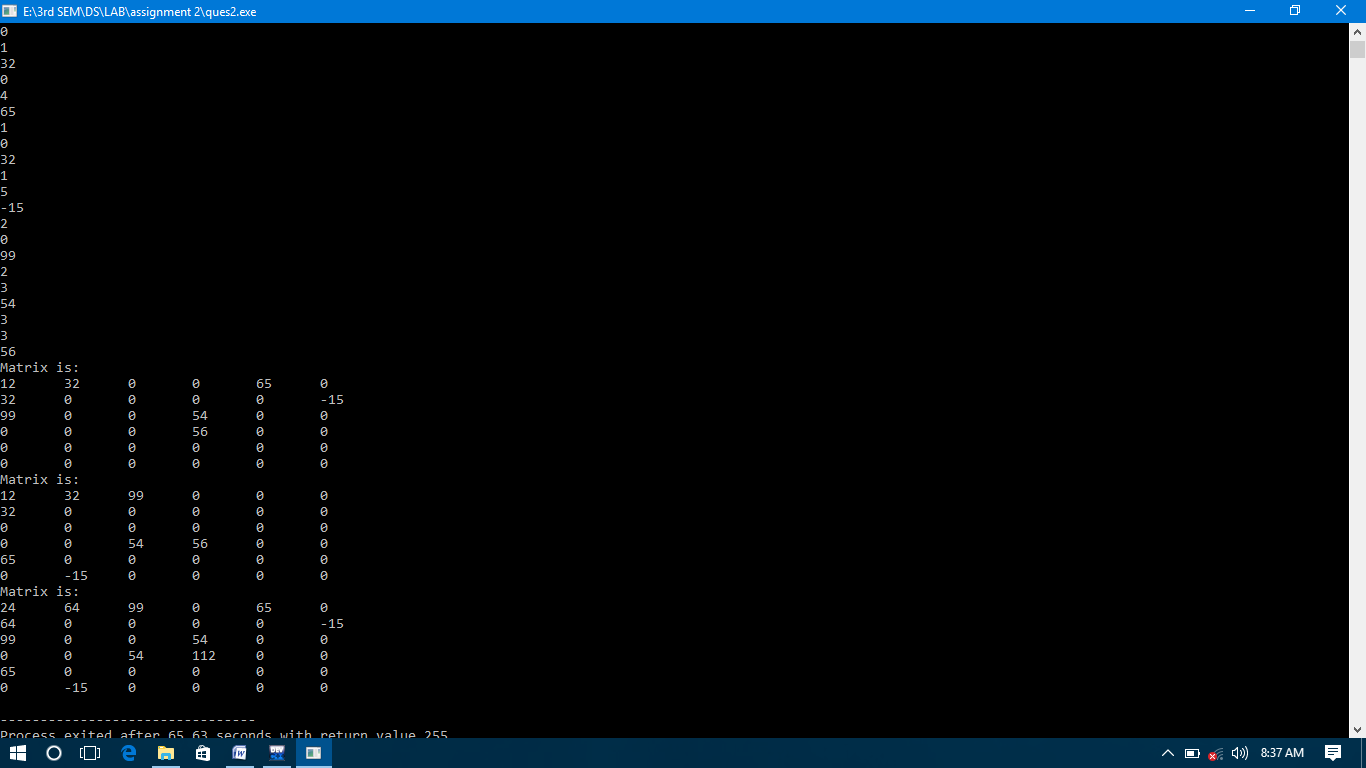
transposesp(a,b);

printsp(b);

addsp(a,b,c);

printsp(c);

}



**Q3**

#include<iostream>

using namespace std;

int main()

{

int a[4][4],i,j,s,n,m;

cout<<"Enter the number of rows: "<<endl;

cin>>m;

cout<<"Enter the number of columns: "<<endl;

cin>>n;

cout<<endl<<"Enter the elements of the 2-D Matrix : ";

for(i=0;i<m;i++)

{

for(j=0;j<n;j++)

{

cout<<"enter a["<<i<<"]["<<j<<"] : "<<endl;

cin>>a[i][j];

}

}

for(i=0;i<m;i++)

{

cout<<endl<<"Sum of elements of row "<<i+1<<" is : ";

s=0;

for(j=0;j<n;j++)

{

s=s+a[i][j];

}

cout<<s;

}

for(j=0;j<n;j++)

{

cout<<endl<<"Sum of elements of column "<<j+1<<" is : ";

s=0;

for(i=0;i<n;i++)

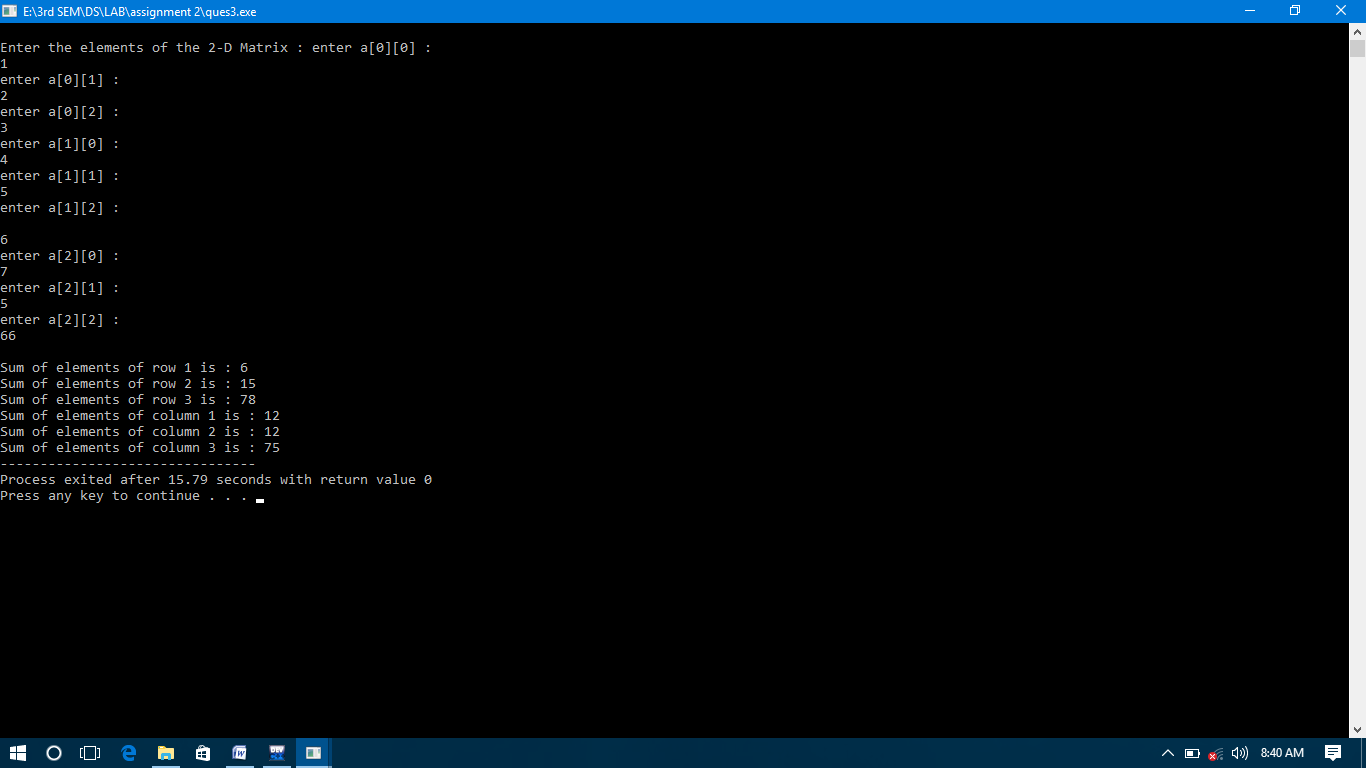
{

s=s+a[i][j];

}

cout<<s;

}

} 

**Q4**

#include<iostream>

using namespace std;

int main()

{

int i,j,r,c;

cout<<"enter no. of rows:"<<endl;

cin>>r;

cout<<"enter no. of columns:"<<endl;

cin>>c;

int a[r][c];

cout<<"Enter Your Matrix:"<<endl;

for(i=0; i<r; i++)

{

for(j=0; j<c; j++)

{

cout<<"enter a["<<i<<"]["<<j<<"] : "<<endl;

cin>>a[i][j];

}

}

for(i=0; i<r; i++)

{

int minr=a[i][0],coli=0;

for(j=1; j<c; j++)

{

if(minr>a[i][j])

{

minr=a[i][j];

coli=j;

}

}

int k;

for (k=0; k<r; k++)

{

if(minr < a[k][coli])

break;

}

if(k==r)

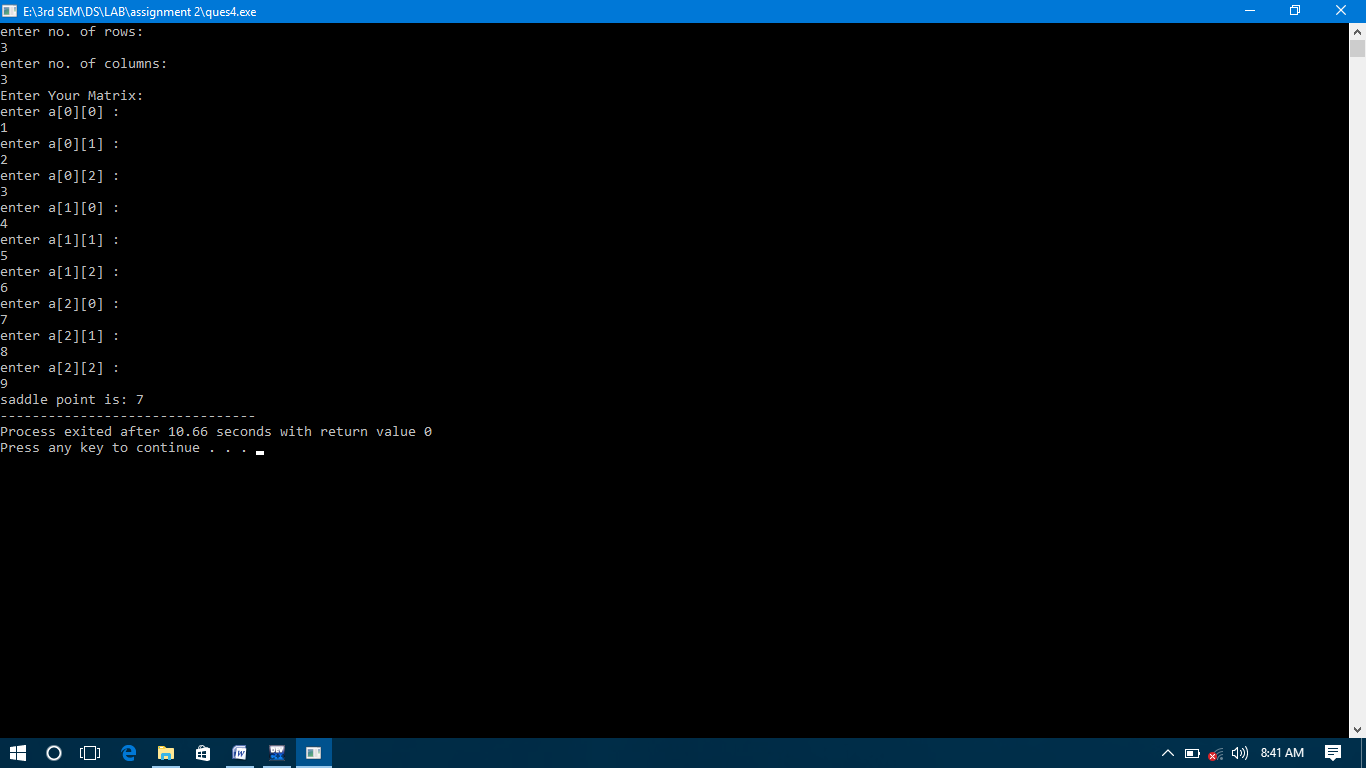
{

cout<<"saddle point is: "<<minr;

}

}

}



**Q5**

#include<iostream>

using namespace std;

int main()

{

int m,n;

cout<<"Enter the number of rows: "<<endl;

cin>>m;

cout<<"Enter the number of columns: "<<endl;

cin>>n;

int i,j,a[m][n];

for(i=0; i<m; i++)

{

for(j=0; j<n; j++)

{

cout<<"enter a["<<i<<"]["<<j<<"] : "<<endl;

cin>>a[i][j];

}

}

int t=0,b=m-1,l=0,r=n-1,dir=0;

while(t<=b && l<=r)

{

if(dir==0)

{

for(i=l; i<=r; i++)

cout<<a[t][i]<<", ";

t++;

}

else if(dir==1)

{

for(i=t; i<=b; i++)

cout<<a[i][r]<<", ";

r--;

}

else if(dir==2)

{

for(i=r; i>=l; i--)

cout<<a[b][i]<<", ";

b--;

}

else if(dir==3)

{

for(i=b; i>=t; i--)

cout<<a[i][l]<<", ";

l++;

}

dir=(dir+1)%4;

}

}