**Hanzla saeed**

**472684**

**Section C**

**Lab manual 9(Lab Tasks)**

**Q#1**

#include<bits/stdc++.h>

using namespace std;

int main()

{ float x1=0,x2=0;

int i,j,a=0,b=0,c=2;

float arr[3][3];

for( i=0;i<=2;i++)

{

for( j=0;j<=2;j++)

{ cout<<"Enter value of matrix:"<<"["<<i<<"]["<<j<<"] :";

cin>>arr[i][j];

cout<<endl;

}

}

cout<<"Mtrix will appear as:"<<endl;

while( a<=2&& b<=2)

{

x1=x1+arr[a][b];

a++;b++;

}

a=0;b=2;

while(a<=2&&b>=0)

{

x2=x2+arr[a][b];

a++;b--;

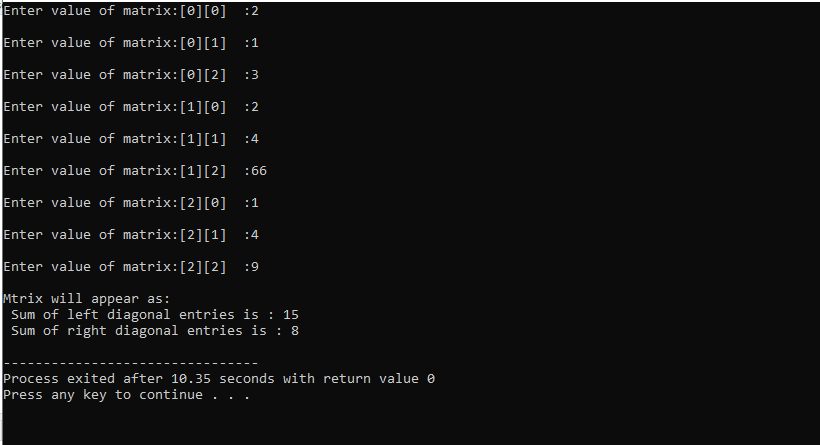
}

cout<<" Sum of left diagonal entries is : "<<x1<<endl;

cout<<" Sum of right diagonal entries is : "<<x2<<endl;

return 0;

}

****

**Q#2**

#include<bits/stdc++.h>

using namespace std;

void function(float arrR[3][3], float arr1[3][3],float arr2[3][3])

{

for( int a=0;a<=2;a++)

{

for(int b=0;b<=2;b++)

{

arrR[a][b]=arr1[a][b]+arr2[a][b];

}

}

}

int main()

{

cout<<endl;

float arr1[3][3],arr2[3][3],arrR[3][3];

int a,b,c,d,i;

for( i=0;i<=2;i++)

{

for( int j=0;j<=2;j++)

{ cout<<" Enter value of first matrix:"<<"["<<i<<"]["<<j<<"] :";

cin>>arr1[i][j];

cout<<endl;

}

}

for(i=0;i<=2;i++) {

for( int j=0;j<=2;j++)

{ cout<<" Enter value of second matrix:"<<"["<<i<<"]["<<j<<"] :";

cin>>arr2[i][j];

cout<<endl;

}

}

function( arrR, arr1,arr2);

for(int x=0;x<=2;x++)

{

for(int y=0;y<=2;y++)

{

cout<<" "<<arrR[x][y]<<" ";

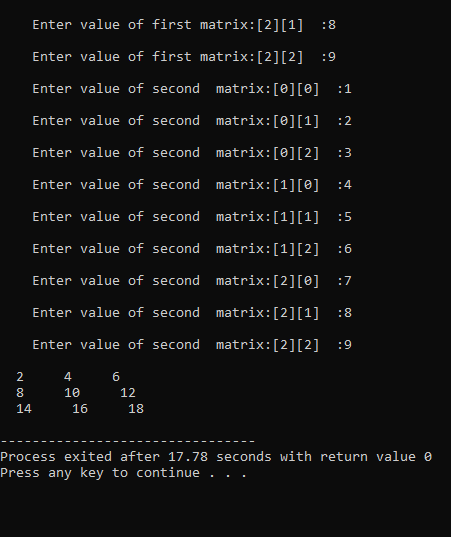
}

cout<<endl;

}

return 0;

}



**Q#3**

#include<bits/stdc++.h>

using namespace std;

void inverse(float arr1[3][3],float arr2[3][3] )

{

for(int a=0;a<=2;a++)

{

for(int b=0;b<=2;b++)

{

arr2[a][b]= arr1[b][a];

}

}

}

int main()

{

float Matrix1[3][3],Matrix2[3][3];

for( int i=0;i<=2;i++)

{

for(int j=0;j<=2;j++)

{ cout<<" Enter value of matrix:"<<"["<<i<<"]["<<j<<"] :";

cin>>Matrix1[i][j];

cout<<endl;

}

}

cout<<" original matrix:-";

cout<<endl;

for( int i=0;i<=2;i++)

{

for(int j=0;j<=2;j++)

{ cout<<" "<<Matrix1[i][j]<<" ";

}

cout<<endl;

}

inverse(Matrix1,Matrix2);

cout<<" Matrix after inverse:-";

cout<<endl;

for( int i=0;i<=2;i++)

{

for(int j=0;j<=2;j++)

{ cout<<" "<<Matrix2[i][j]<<" ";

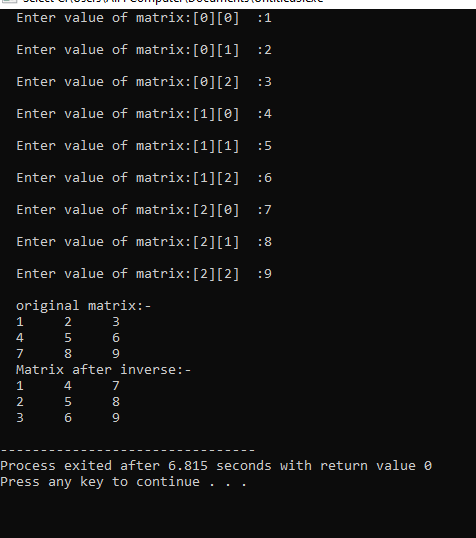
}

cout<<endl;

}

return 0;

}



**Q#4**

#include<bits/stdc++.h>

using namespace std;

void function(float arrR[3][3], float arr1[3][3],float arr2[3][3])

{

for(int a = 0; a < 3; a++)

{

for(int b = 0; b < 3; b++)

{

arrR[a][b] = 0;

}

}

for(int a = 0; a < 3; a++)

{

for(int b = 0; b < 3; b++)

{

for(int c = 0; c < 3; c++)

{

arrR[a][b] += arr1[a][c] \* arr2[c][b];

}

}

}

}

int main()

{

cout<<endl;

float arr1[3][3],arr2[3][3],arrR[3][3];

int a,b,c,d,i;

for( i=0;i<=2;i++)

{

for( int j=0;j<=2;j++)

{ cout<<" Enter value of first matrix:"<<"["<<i<<"]["<<j<<"] :";

cin>>arr1[i][j];

cout<<endl;

}

}

for(i=0;i<=2;i++) {

for( int j=0;j<=2;j++)

{ cout<<" Enter value of second matrix:"<<"["<<i<<"]["<<j<<"] :";

cin>>arr2[i][j];

cout<<endl;

}

}

function( arrR, arr1,arr2);

cout<<"First Matrix :-"<<endl;

for( int i=0;i<=2;i++)

{

for(int j=0;j<=2;j++)

{ cout<<" "<<arr1[i][j]<<" ";

}

cout<<endl;

}

cout<<"second Matrix :-"<<endl;

for( int i=0;i<=2;i++)

{

for(int j=0;j<=2;j++)

{ cout<<" "<<arr2[i][j]<<" ";

}

cout<<endl;

}

cout<<"Final matrix after multiplying these two matrixes :-"<<endl;

for(int x=0;x<=2;x++)

{

for(int y=0;y<=2;y++)

{

cout<<" "<<arrR[x][y]<<" ";

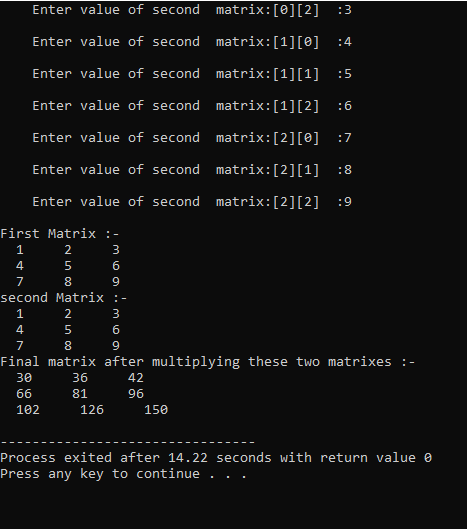
}

cout<<endl;

}

return 0;

}



**Q#5**

#include <bits/stdc++.h>

using namespace std;

int Table(int num, int i = 1)

{

if(i > 10){return num;}

cout << num << " \* " << i << " = " << num \* i <<endl;

Table(num, i + 1);

}

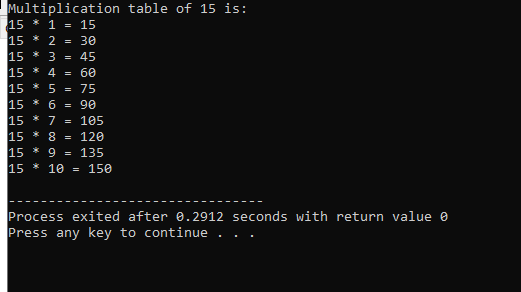
int main() {

cout << "Multiplication table of 15 is:"<<endl;;

Table(15);

return 0;

}

****

**Lab manual 9(Home Task)**

**Q#1**

#include <bits/stdc++.h>

using namespace std;

double determinant(double matrix[3][3]) {

return matrix[0][0] \* (matrix[1][1] \* matrix[2][2] - matrix[1][2] \* matrix[2][1]) -

matrix[0][1] \* (matrix[1][0] \* matrix[2][2] - matrix[1][2] \* matrix[2][0]) +

matrix[0][2] \* (matrix[1][0] \* matrix[2][1] - matrix[1][1] \* matrix[2][0]);

}

void adjoint(double matrix[3][3], double adj[3][3]) {

for (int i = 0; i < 3; ++i) {

for (int j = 0; j < 3; ++j) {

int sign = ((i + j) % 2 == 0) ? 1 : -1;

adj[j][i] = sign \* (matrix[(i + 1) % 3][(j + 1) % 3] \* matrix[(i + 2) % 3][(j + 2) % 3] -

matrix[(i + 1) % 3][(j + 2) % 3] \* matrix[(i + 2) % 3][(j + 1) % 3]);

}

}

}

bool inverse(double matrix[3][3], double inv[3][3]) {

double det = determinant(matrix);

if (det == 0) {

cout << "Inverse cannot be calculated as the determinant is 0." << endl;

return false;

}

double adj[3][3];

adjoint(matrix, adj);

for (int i = 0; i < 3; ++i) {

for (int j = 0; j < 3; ++j) {

inv[i][j] = adj[i][j] / det;

}

}

return true;

}

int main() {

double matrix[3][3];

double inv[3][3];

cout << "Enter the elements of the 3x3 matrix :-" << endl;

for (int i = 0; i < 3; ++i) {

for (int j = 0; j < 3; ++j) {

cout << "Enter the element of the matrix["<<i<<"]["<<j<<"] :";

cin >> matrix[i][j];

}

}

if (inverse(matrix, inv)) {

cout << "The inverse of the matrix is:" << endl;

for (int i = 0; i < 3; ++i) {

for (int j = 0; j < 3; ++j) {

cout << inv[i][j] << " ";

}

cout << endl;

}

}

return 0;

}

