```
#include <windows.h>
#include<GL/glut.h>
#include<math.h>
#include<stdio.h>
#include<iostream>

void display();
using namespace std;
float xmin = -100;
float ymin = -100;
float xmax = 100;
float ymax = 100;
float xd1, yd1, xd2, yd2;



void init(void)
{

	glClearColor(0.0, 0, 0, 0);
	glMatrixMode(GL_PROJECTION);
	gluOrtho2D(-300, 300, -300, 300);

}

int code(float x, float y)
{
	int c = 0;
	if (y > ymax)c = 8;
	if (y < ymin)c = 4;
```

```
        if (x > xmax)c = c | 2;
        if (x < xmin)c = c | 1;
        return c;
}

void cohen_Line(float x1, float y1, float x2, float y2)
{
        int c1 = code(x1, y1);
        int c2 = code(x2, y2);
        float m = (y2 - y1) / (x2 - x1);
        while ((c1 | c2) > 0)
        {
                if ((c1 & c2) > 0)
                {
                        exit(0);
                }

                float xi = x1; float yi = y1;
                int c = c1;
                if (c == 0)
                {
                        c = c2;
                        xi = x2;
                        yi = y2;
                }
                float x, y;
                if ((c & 8) > 0)
                {
                        y = ymax;
```

```
        x = xi + 1.0 / m * (ymax - yi);
}
else
        if ((c & 4) > 0)
        {
                y = ymin;
                x = xi + 1.0 / m * (ymin - yi);
        }
        else
                if ((c & 2) > 0)
                {
                        x = xmax;
                        y = yi + m * (xmax - xi);
                }
                else
                        if ((c & 1) > 0)
                        {
                                x = xmin;
                                y = yi + m * (xmin - xi);
                        }

if (c == c1)
{
        xd1 = x;
        yd1 = y;
        c1 = code(xd1, yd1);
}

if (c == c2)
```

```
		{
			xd2 = x;
			yd2 = y;
			c2 = code(xd2, yd2);
		}
	}

	display();

}

void mykey(unsigned char key, int x, int y)
{
	if (key == 'c')
	{
		cout << "Hello";
		cohen_Line(xd1, yd1, xd2, yd2);
		glFlush();
	}
}
void display()
{

	glClear(GL_COLOR_BUFFER_BIT);
	glColor3f(0.0, 1.0, 0.0);

	glBegin(GL_LINE_LOOP);
	glVertex2i(xmin, ymin);
	glVertex2i(xmin, ymax);
	glVertex2i(xmax, ymax);
```

```
        glVertex2i(xmax, ymin);
        glEnd();
        glColor3f(1.0, 0.0, 0.0);
        glBegin(GL_LINES);
        glVertex2i(xd1, yd1);
        glVertex2i(xd2, yd2);
        glEnd();
        glFlush();



}



int main(int argc, char** argv)
{
        printf("Enter line co-ordinates:");
        cin >> xd1 >> yd1 >> xd2 >> yd2;
        glutInit(&argc, argv);
        glutInitDisplayMode(GLUT_SINGLE | GLUT_RGB);
        glutInitWindowSize(600, 600);
        glutInitWindowPosition(0, 0);
        glutCreateWindow("Clipping");
        glutDisplayFunc(display);
        glutKeyboardFunc(mykey);
        init();
        glutMainLoop();
        return 0;
}
```