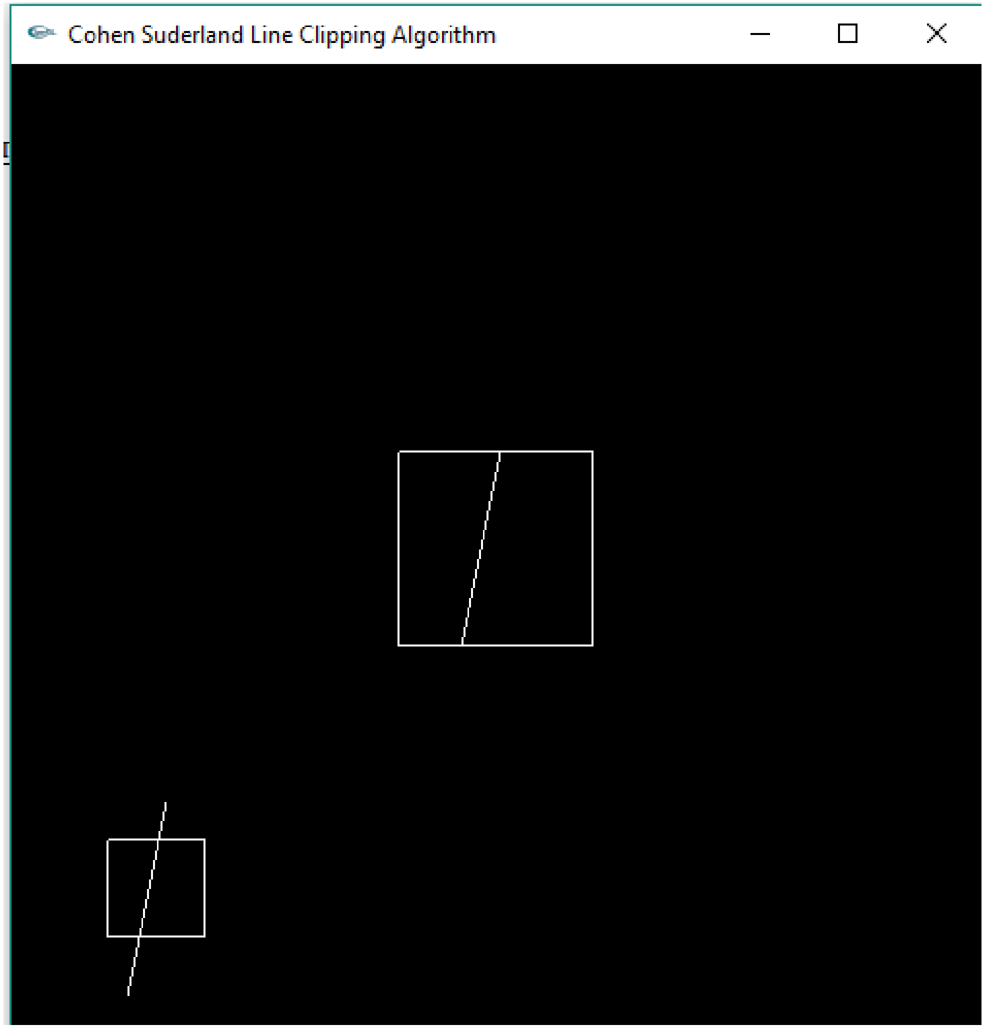
Program to clip a lines using Cohen-Sutherland line-clipping algorithm.

```
#include <stdio.h>
#include <GL\glut.h>

double xmin=50,ymin=50, xmax=100,ymax=100;
double xvmin=200,yvmin=200,xvmax=300,yvmax=300;
const int RIGHT = 8;
const int LEFT = 2;
const int TOP = 4;
const int BOTTOM = 1;

int ComputeOutCode (double x, double y)
{
 int code = 0;
 if (y > ymax)              //above the clip window
  code |= TOP;
 else if (y < ymin)         //below the clip window
  code |= BOTTOM;
 if (x > xmax)              //to the right of clip window
  code |= RIGHT;
 else if (x < xmin)         //to the left of clip window
  code |= LEFT;
 return code;
}
void CohenSutherland(double x0, double y0,double x1, double y1)
{
 int outcode0, outcode1, outcodeOut;
bool accept = false, done = false;
outcode0 = ComputeOutCode (x0, y0);
outcode1 = ComputeOutCode (x1, y1);
do{
  if (!(outcode0 | outcode1))
  {
   accept = true;
   done = true;
  }
  else if (outcode0 & outcode1)
   done = true;
  else {
   double x, y;
   outcodeOut = outcode0? outcode0: outcode1;
   if (outcodeOut & TOP)
   {
    x = x0 + (x1 - x0) * (ymax - y0)/(y1 - y0);
    y = ymax;
   }
   else if (outcodeOut & BOTTOM)
   {
    x = x0 + (x1 - x0) * (ymin - y0)/(y1 - y0);
    y = ymin;
   }
   else if (outcodeOut & RIGHT)
   {
    y = y0 + (y1 - y0) * (xmax - x0)/(x1 - x0);
    x = xmax;
   }
   else
   {
    y = y0 + (y1 - y0) * (xmin - x0)/(x1 - x0);
    x = xmin;
   }

if (outcodeOut == outcode0)
   {
    x0 = x;
    y0 = y;
    outcode0 = ComputeOutCode (x0, y0);
   }
   else
   {
    x1 = x;
    y1 = y;
    outcode1 = ComputeOutCode (x1, y1);
   }
  }
 }while (!done);

 if (accept)
 {
double sx=(xvmax-xvmin)/(xmax-xmin);
  double sy=(yvmax-yvmin)/(ymax-ymin);
  double vx0=xvmin+(x0-xmin)*sx;
  double vy0=yvmin+(y0-ymin)*sy;
  double  vx1=xvmin+(x1-xmin)*sx;
  double vy1=yvmin+(y1-ymin)*sy;

  glColor3f(1.0, 1.0, 1.0);
  glBegin(GL_LINE_LOOP);
   glVertex2f(xvmin, yvmin);
   glVertex2f(xvmax,  yvmin);
  glVertex2f(xvmax,  yvmax);
  glVertex2f(xvmin,   yvmax);
  glEnd();

  glColor3f(1.0,1.0,1.0);
  glBegin(GL_LINES);
   glVertex2d (vx0, vy0);
  glVertex2d  (vx1, vy1);
  glEnd();
 }
}

void display()
{
double x0=60,y0=20,x1=80,y1=120;
glClear(GL_COLOR_BUFFER_BIT);
glColor3f(1.0,1.0,1.0);
glBegin(GL_LINES);
   glVertex2d (x0, y0);
   glVertex2d (x1, y1);

glEnd();
glColor3f(1.0, 1.0, 1.0);
glBegin(GL_LINE_LOOP);
    glVertex2f(xmin,   ymin);
    glVertex2f(xmax,  ymin);
    glVertex2f(xmax, ymax);
    glVertex2f(xmin, ymax);
}

glEnd();
CohenSutherland(x0,y0,x1,y1);
glFlush();
void myinit()
{
 glClearColor(0.0,0.0,0.0,1.0);
 glMatrixMode(GL_PROJECTION);
 glLoadIdentity();
 gluOrtho2D(0.0,500.0,0.0,500.0);
 glMatrixMode(GL_MODELVIEW);

}
void main(int argc, char **argv)
{
 glutInit(&argc,argv);
 glutInitDisplayMode(GLUT_SINGLE|
 GLUT_RGB);
 glutInitWindowSize(500,500);
 glutInitWindowPosition(0,0);
 glutCreateWindow("Cohen Suderland Line Clipping
 Algorithm"); myinit();
glutDisplayF
 unc(display);
 glutMainLoo
 p();
}
```

OUTPUT :


Cohen Suderland Line Clipping Algorithm