Code: -

```
#include<windows.h>
#include<GL/glu.h>
#include<GL/glut.h>

GLfloat xMin=-0.5,xMax=0.5,yMin=-0.5,yMax=0.5;
//GLfloat x1=-0.8,y1=-0.6,x2=0.7,y2=0.4; // -> Partially Inside
//GLfloat x1=-0.7,y1=-0.3,x2=-0.6,y2=0.9; // -> Completely Outside
GLfloat x1=-0.4,y1=-0.3,x2=0.2,y2=0.4; // -> Completely Inside

int Left=1,Right=2,Bot=4,Top=8;
int C1,C2;
int Clip_Flag = 0, Flag = 1;;


int Get_Code(GLfloat x,GLfloat y)
{
   int Code = 0;
   if(x<xMin)
      Code = Code | Left;
   if(x>xMax)
      Code = Code | Right;
   if(y<yMin)
      Code = Code | Bot;
   if(y>yMax)
      Code = Code | Top;
```

```
    return Code;
}

void Clip()
{
    int C;
    GLfloat x,y;
    if(C1)
        C = C1;
    else
        C = C2;

    if(C & Left)
    {
        x = xMin;
        y = y1+(y2-y1)*((xMin-x1)/(x2-x1));
    }
    if(C & Right)
    {
        x = xMax;
        y = y1+(y2-y1)*((xMax-x1)/(x2-x1));
    }
    if(C & Bot)
    {
        y = yMin;
        x = x1+(x2-x1)*((yMin-y1)/(y2-y1));
```

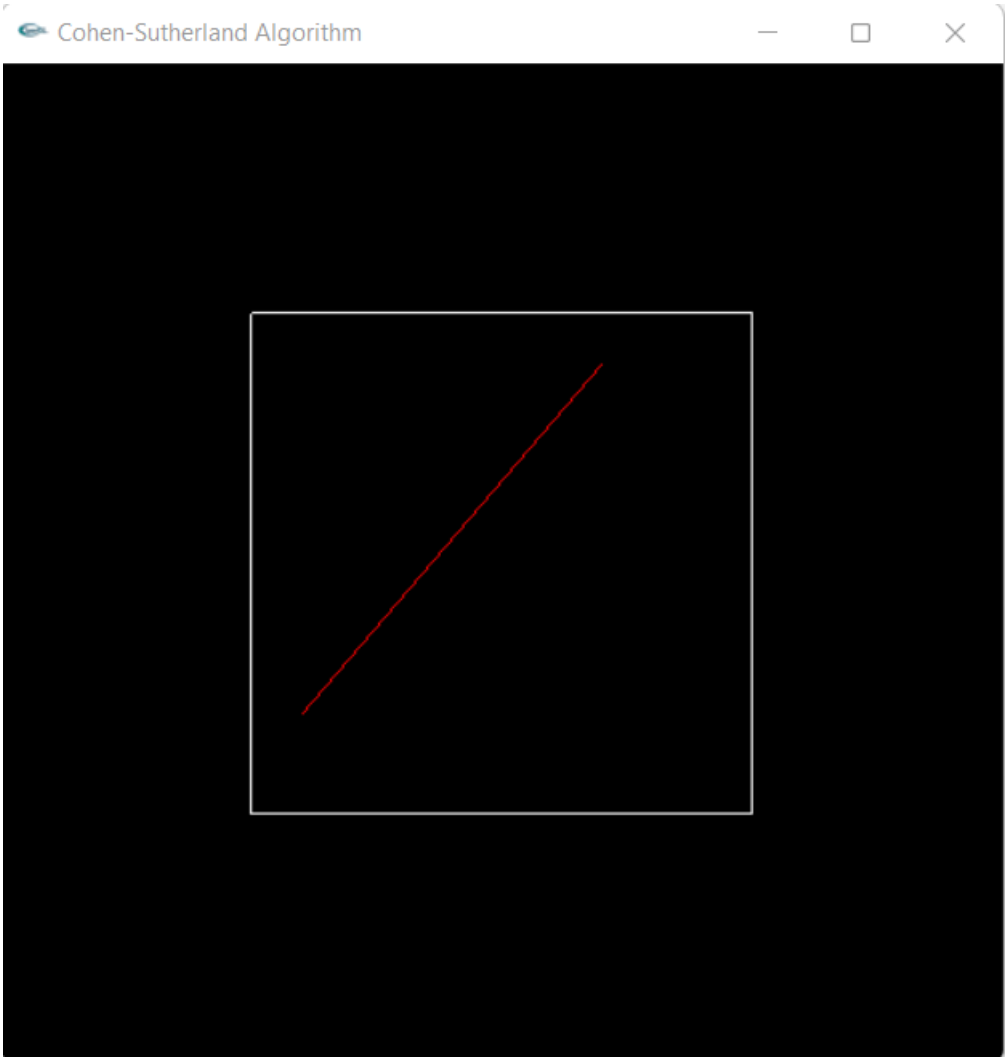
```
    }
    if(C & Top)
    {
        y = yMax;
        x = x1+(x2-x1)*((yMax-y1)/(y2-y1));
    }

    if(C == C1)
    {
        x1 = x;
        y1 = y;
    }
    else
    {
        x2 = x;
        y2 = y;
    }
}

void Draw()
{
    glClear(GL_COLOR_BUFFER_BIT);

    glColor3f(1,1,1);
    glBegin(GL_LINE_LOOP);
        glVertex2f(xMin,yMin);
```

```
   glVertex2f(xMax,yMin);
   glVertex2f(xMax,yMax);
   glVertex2f(xMin,yMax);
glEnd();

glColor3f(1,0,0);
if(Flag == 1)
{
 glBegin(GL_LINES);
   glVertex2f(x1,y1);
   glVertex2f(x2,y2);
glEnd();
}

while(1 & Clip_Flag == 1)
{
   C1 = Get_Code(x1,y1);
   C2 = Get_Code(x2,y2);

   if((C1|C2) == 0)
      break;
   else if((C1&C2)!=0)
   {
      Flag = 0;
      break;
   }
```

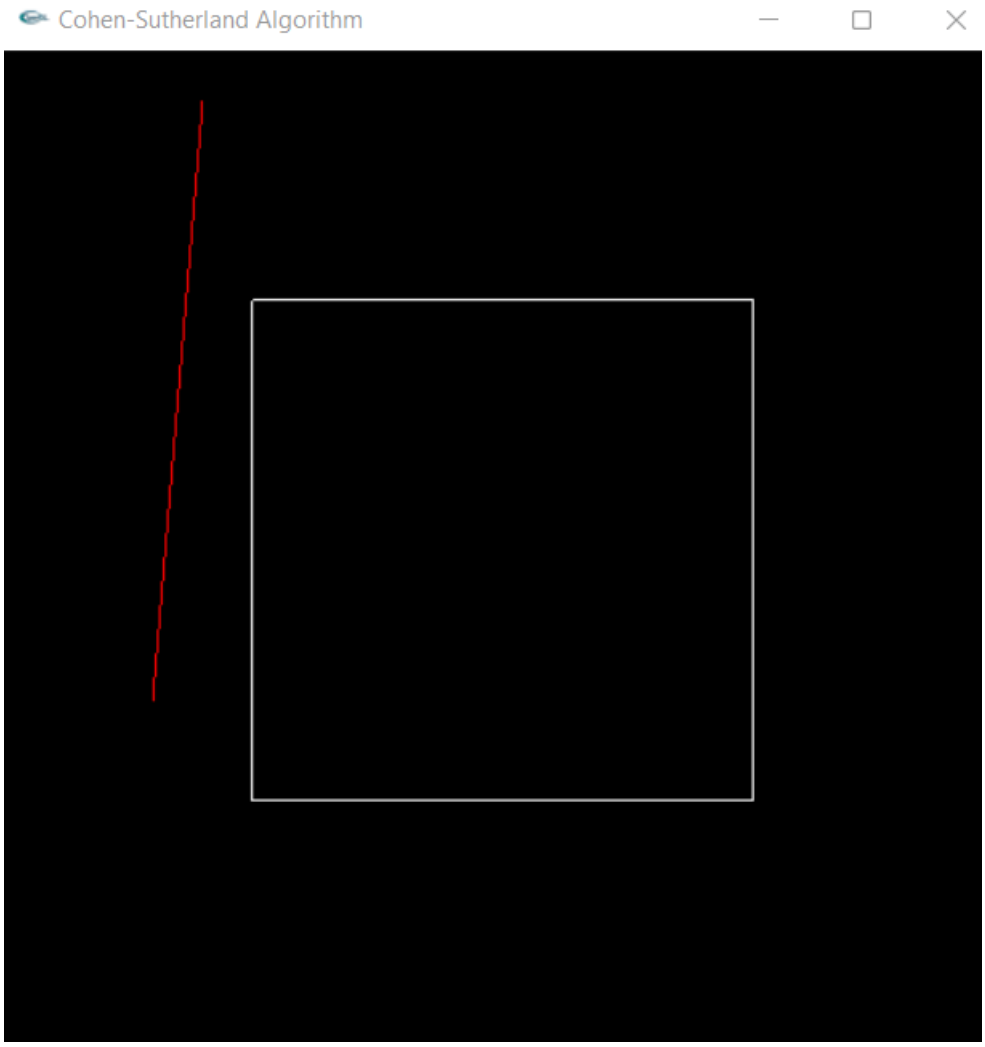
```
        else
            Clip();
    }
    glFlush();
}

void Key(unsigned char ch,int x,int y)
{
    Clip_Flag = 1;
    glutPostRedisplay();
}

int main(int argC,char *argV[])
{
    glutInit(&argC,argV);
    glutInitWindowSize(500,500);
    glutInitWindowPosition(100,100);
    glutInitDisplayMode(GLUT_RGB | GLUT_SINGLE);
    glutCreateWindow("Cohen-Sutherland Algorithm");
    glutDisplayFunc(Draw);
    glutKeyboardFunc(Key);
    glutMainLoop();
    return 0;

}
```

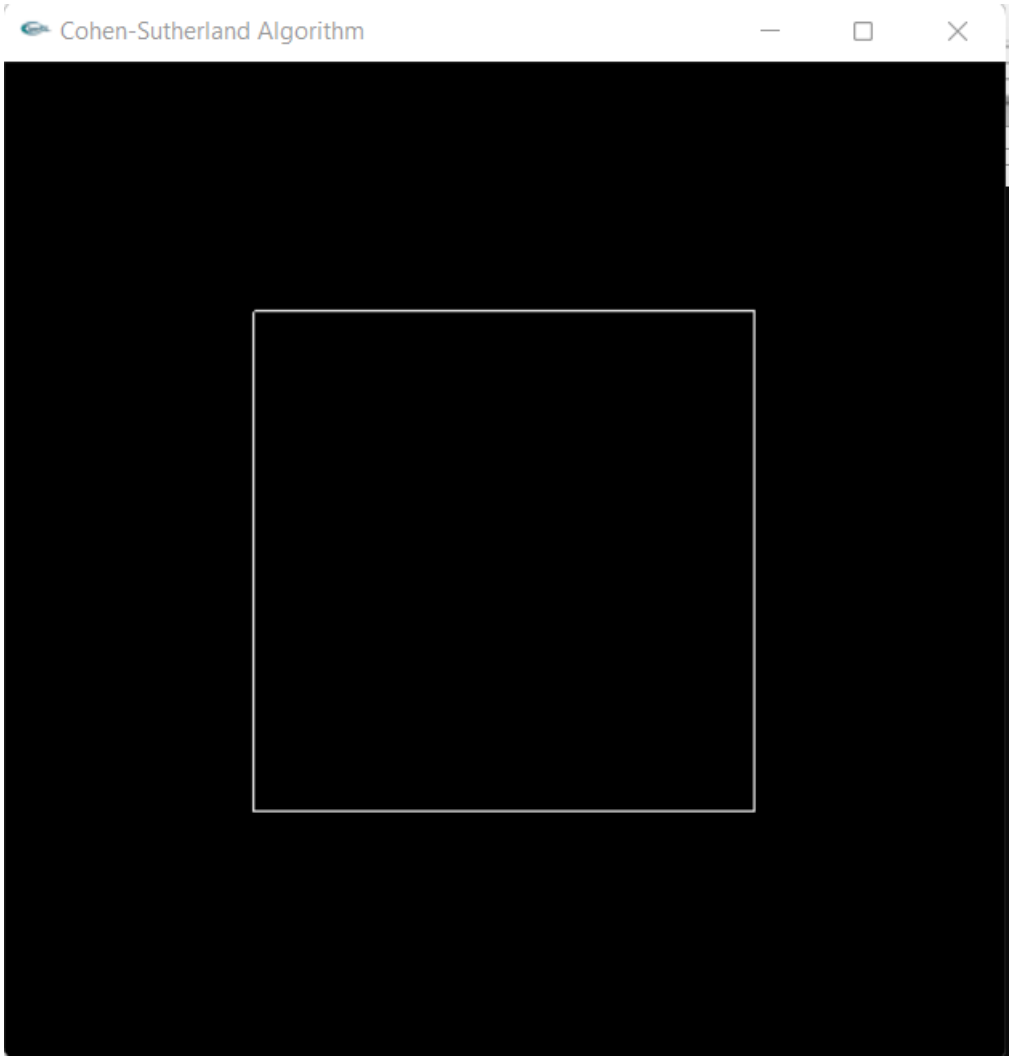
Output: -

i) Completely Inside

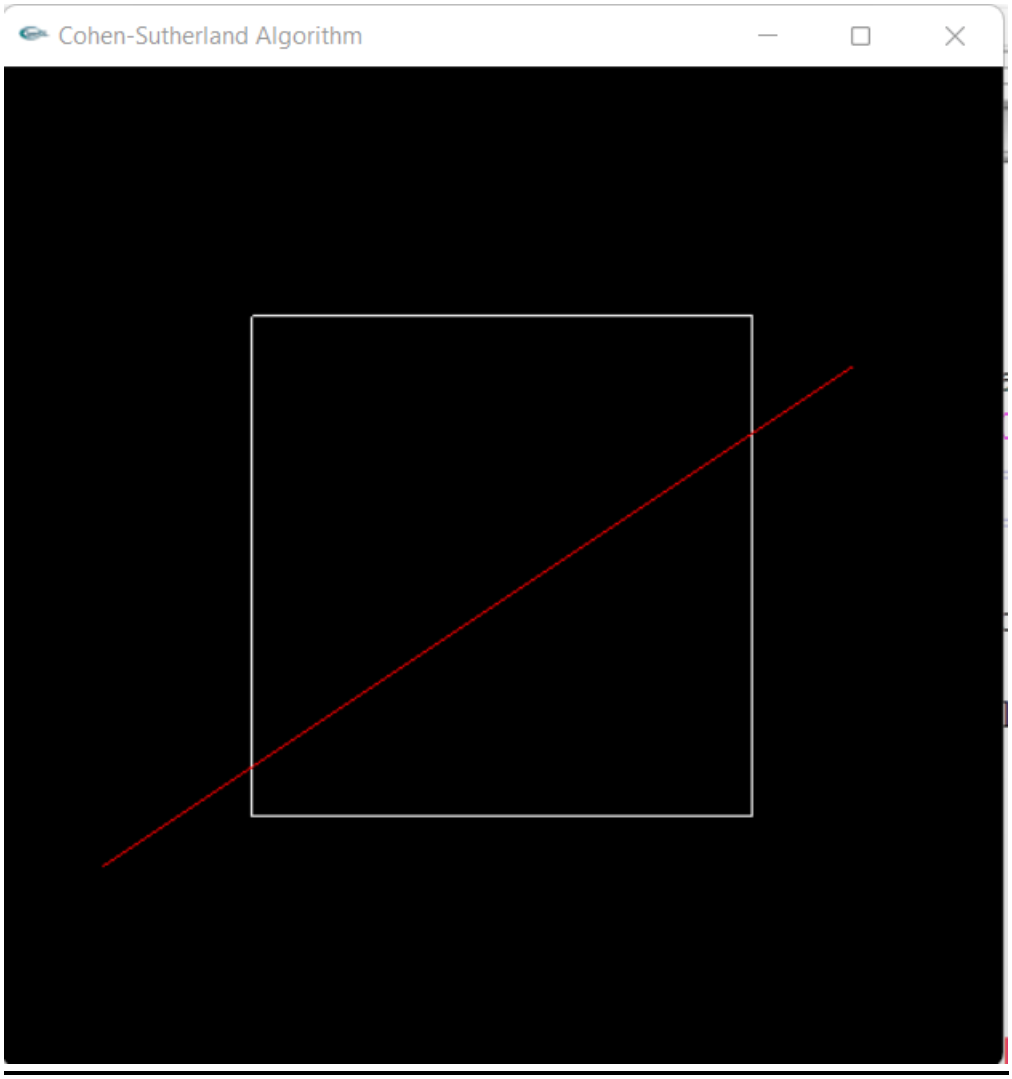


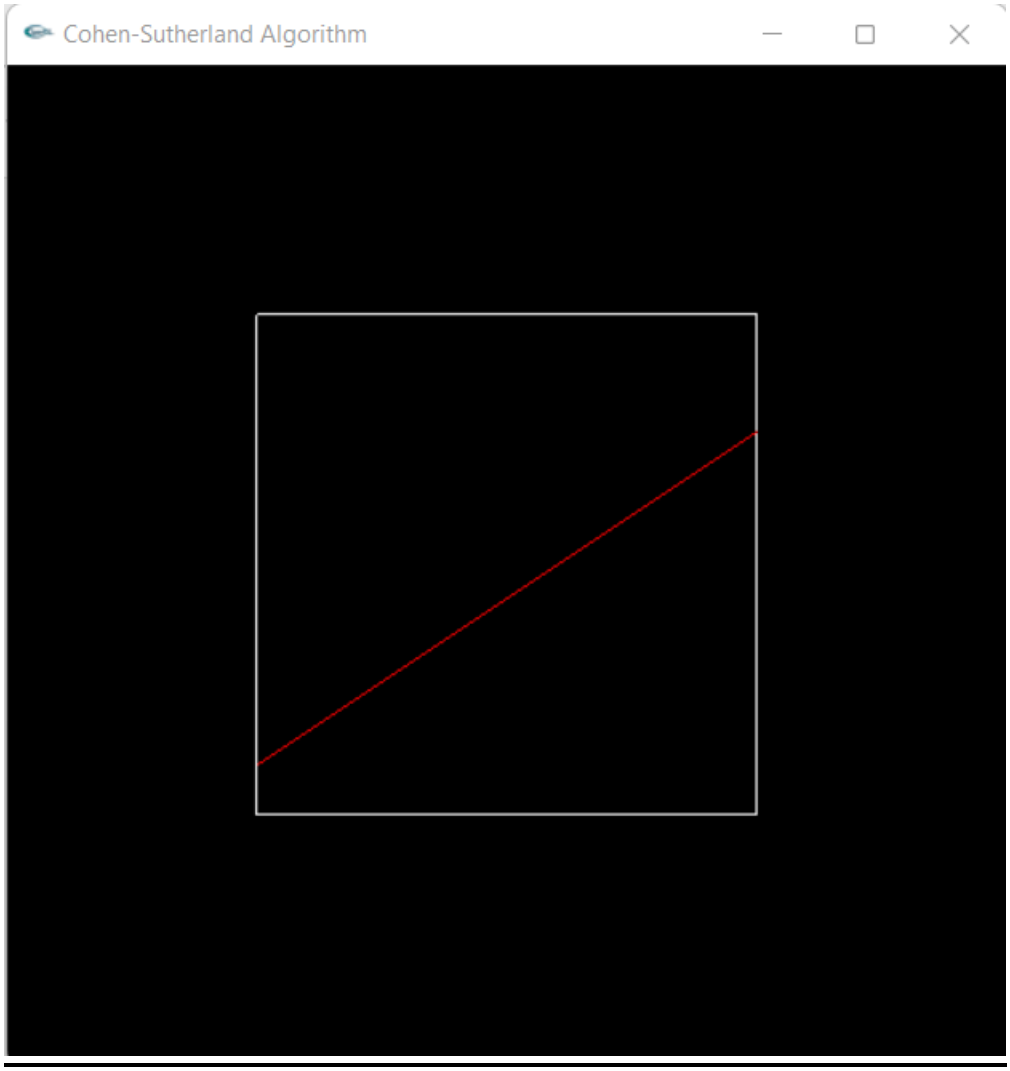
ii) Completely Outside


Cohen-Sutherland Algorithm



i) Partially Inside


Cohen-Sutherland Algorithm


Cohen-Sutherland Algorithm