MALNAD COLLEGE OF ENGINEERING

(An Autonomous Institute under VTU, Belagavi)

HASSAN – 573202

Subject: Computer Graphics and Visualization **Code:**

18CS602

Submitted by:

Jayshankar

4MC18CS057

6th Sem, B sec

Computer Science

Liang barsky program

#include <stdio.h>

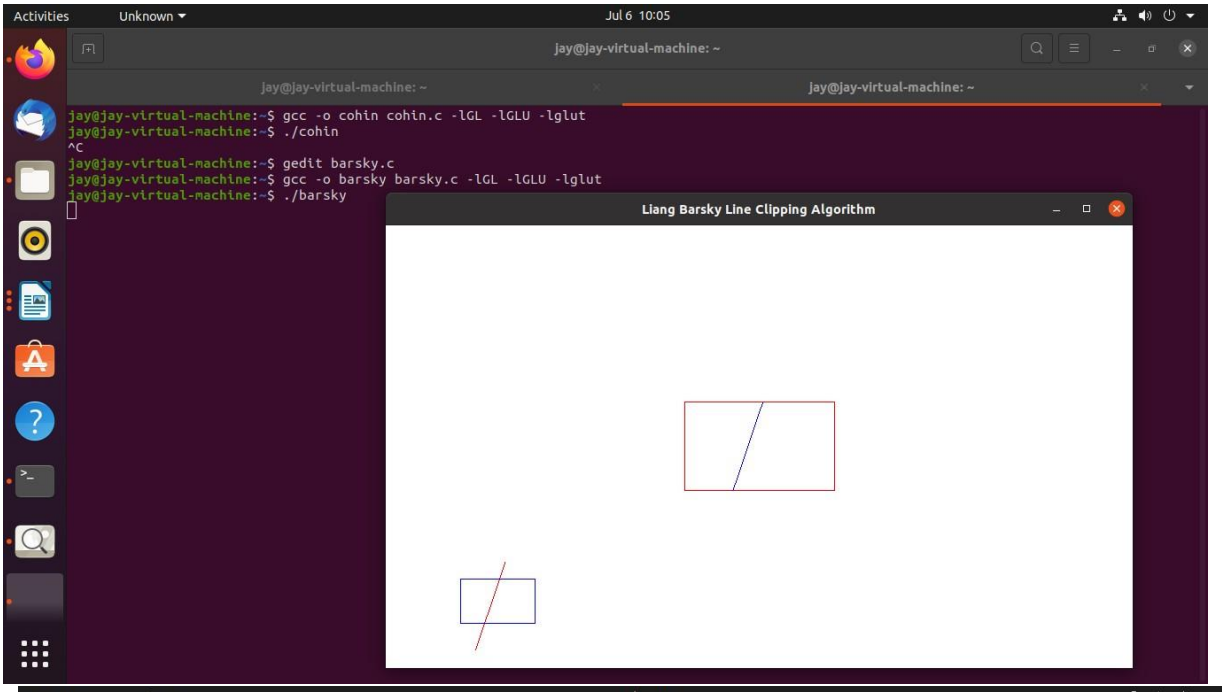
```
#include &lt;GL/glut.h&gt;
#define true 1 #define
false 0
double xmin=50,ymin=50, xmax=100,ymax=100; // Window boundaries double
xvmin=200,yvmin=200,xvmax=300,yvmax=300; // Viewport boundaries
//int x1, x2, y1, y2; int cliptest(double p, double q,
double *t1, double *t2)
{
double t; if(p) t=q/p; // to check whether p if(p
&lt; 0.0) // potentially entry point, update te
{
if( t &gt; *t1) *t1=t; if( t &gt; *t2) return(false); //
line portion is outside
} else if(p &gt; 0.0) // Potentially leaving point,
update tl
{
if( t &lt; *t2) *t2=t; if( t &lt; *t1) return(false); //
line portion is outside
} else if(p
== 0.0)
{
if( q &lt; 0.0) return(false); // line parallel to edge but outside
}
return(true);
}
void LiangBarskyLineClipAndDraw (double x0, double y0,double x1, double y1)
{
```

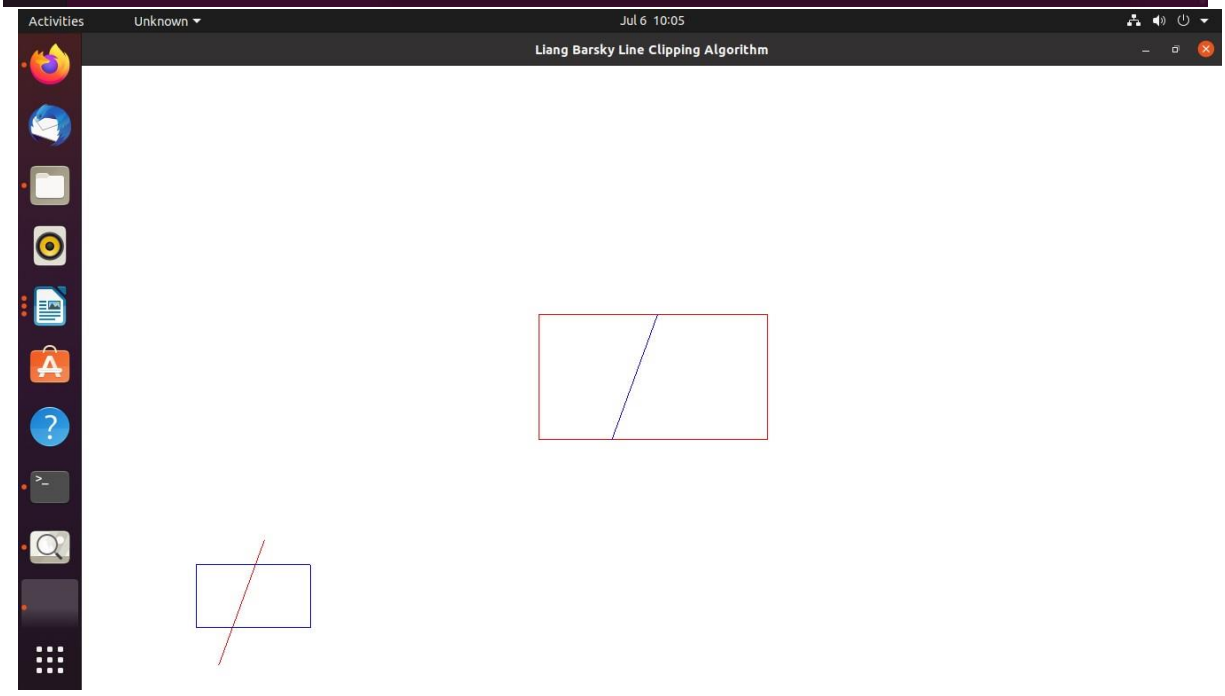
```
double dx=x1-x0, dy=y1-y0, te=0.0, tl=1.0; if(cliptest(-dx,x0-xmin,&amp;amp;te,&amp;amp;tl))
// inside test wrt left edge if(cliptest(dx,xmax-x0,&amp;amp;te,&amp;amp;tl)) // inside test wrt
right edge if(cliptest(-dy,y0-ymin,&amp;amp;te,&amp;amp;tl)) // inside test wrt bottom edge
if(cliptest(dy,ymax-y0,&amp;amp;te,&amp;amp;tl)) // inside test wrt top edge
{ if( tl &amp;lt;
1.0 )
{
x1 = x0 + tl*dx; y1
= y0 + tl*dy;
}
if( te &amp;gt; 0.0 ) {
x0 = x0 + te*dx;
y0 = y0 + te*dy;
}

// Window to viewport mappings double sx=(xvmax-
xvmin)/(xmax-xmin); // Scale parameters 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; //draw
a red colored viewport glColor3f(1.0,
0.0, 0.0);
glBegin(GL_LINE_LOOP);
glVertex2f(xvmin, yvmin); glVertex2f(xvmax, yvmin);
glVertex2f(xvmax, yvmax); glVertex2f(xvmin,
```

```
yvmax); glEnd(); glColor3f(0.0,0.0,1.0); // draw blue
colored clipped line
glBegin(GL_LINES);
glVertex2d (vx0, vy0);
glVertex2d (vx1, vy1); glEnd();
}
}// end of line clipping void
display()
{
double x0=60,y0=20,x1=80,y1=120;
glClear(GL_COLOR_BUFFER_BIT);
//draw the line with red color glColor3f(1.0,0.0,0.0);
glBegin(GL_LINES);
glVertex2d (x0, y0);
glVertex2d (x1, y1); glEnd();
//draw a blue colored window
glColor3f(0.0, 0.0, 1.0);
glBegin(GL_LINE_LOOP);
glVertex2f(xmin, ymin);
glVertex2f(xmax, ymin);
glVertex2f(xmax, ymax);
glVertex2f(xmin, ymax); glEnd();
LiangBarskyLineClipAndDraw(x0,y0,x1,y1);
glFlush();
}
void myinit()
{
```

```
glClearColor(1.0,1.0,1.0,1.0);
glColor3f(1.0,0.0,0.0); glPointSize(1.0);

glMatrixMode(GL_PROJECTION);
glLoadIdentity(); gluOrtho2D(0.0,499.0,0.0,499.0);
}
int main(int argc, char** argv)
{
//printf(&amp;quot;Enter End points:&amp;quot;);
//scanf(&amp;quot;%d%d%d%d&amp;quot;, &amp;amp;x1,&amp;amp;x2,&amp;amp;y1,&amp;amp;y2);
glutInit(&amp;amp;argc,argv);
glutInitDisplayMode(GLUT_SINGLE|GLUT_RGB);
glutInitWindowSize(500,500);
glutInitWindowPosition(0,0); glutCreateWindow(&amp;quot;Liang Barsky
Line Clipping Algorithm&amp;quot;); glutDisplayFunc(display); myinit();
glutMainLoop();
}
```

Output:


Activities
Unknown
Jul 6 10:05
jay@jay-virtual-machine: ~
jay@jay-virtual-machine: ~
jay@jay-virtual-machine: ~
jay@jay-virtual-machine:~$ gcc -o cohin cohin.c -lGL -lGLU -lglut
jay@jay-virtual-machine:~$ ./cohin
^C
jay@jay-virtual-machine:~$ gedit barsky.c
jay@jay-virtual-machine:~$ gcc -o barsky barsky.c -lGL -lGLU -lglut
jay@jay-virtual-machine:~$ ./barsky
Liang Barsky Line Clipping Algorithm


Activities
Unknown
Jul 6 10:05
Liang Barsky Line Clipping Algorithm