```
#include<stdio.h>
#include<math.h>
#include<GL/glut.h>
#include<GL/glu.h>
#include<GL/gl.h>

int X1 , Y1 , X2 , Y2 ;

  int round_value(float p)
{
   int q=p;
    if((p-q)>0.5)
       return q+1;
    else
       return q;
}

void LineBA(int x1,int y1,int x2,int y2)
{
int dx=(X2-X1);
int dy=(Y2-Y1);
int e = 2*dy-dx;
int length;
float x=X1,y=Y1;

if(abs(dy)>abs(dx))
{
length=abs(dy);
}
else
{
length=abs(dx);
}

glBegin(GL_POINTS);
glVertex2d(x,y);

int i;
for(i=0;i<=length;i++)
{
if(e<0)
{
 x=x+1;
 e= e+2*dy;
 }
 else
 {
  x=x+1;
  y=y+1;
  e = e - 2*dx;
```

```
 }
 glVertex2d(x,y);
}
 glEnd();
 glFlush();
}

void display()
{
LineBA( X1 , X2 , Y1 , Y2 );

}
void Init()
{
glClearColor(1.0,1.0,1.0,0);
glColor3f(1.0,0.0,0.0);
glClear(GL_COLOR_BUFFER_BIT);
gluOrtho2D(0,640,0,480);
}

void main(int argc, char **argv)
{
printf("/n Enter two endpoints of line to be drawn: ");
printf("/n Enter point1(X1,Y1): ");
scanf("%d%d",&X1,&Y1);
printf("/n Enter point2(X2,Y2): ");
scanf("%d%d",&X2,&Y2);

glutInit(&argc,argv);
glutInitDisplayMode(GLUT_SINGLE | GLUT_RGB);
glutInitWindowPosition(0,0);
glutInitWindowSize(640,480);
glutCreateWindow("BA_LINE");

Init();
glutDisplayFunc(display);
glutMainLoop();
}
```