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



int   x1=0, x2=120, y1=0, y2=70, dx, dy;
float ix, iy,step;


void display(void)
{


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


dx = abs(x2-x1);
dy = abs(y2-y1);


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

```
    {
       step = abs(dy);
    }

    ix = dx/step;
    iy = dy/step;

    float x=x1, y=y1;

    glBegin(GL_POINTS);
    glVertex2i (abs(x), abs(y));

    //glEnd();


    int i;
    for (i = 0; i<step; i++)
    {
       x = x + ix;
       y = y + iy;

       printf("%.1f %.1f %.1f %.1f %.1f", x, y, step, ix, iy);
       printf("\n");

       glVertex2i (abs(x), abs(y));    }

glEnd();
```

```
glutSwapBuffers();
}
void init (void)
{

glClearColor (0.0, 0.0, 0.0, 0.0);
glMatrixMode(GL_PROJECTION);
glLoadIdentity();
glOrtho(0.0, 200.0, 0.0, 200.0, -200.0, 200.0);

}



int main(int argc, char** argv)
{




glutInit(&argc, argv);
glutInitDisplayMode (GLUT_DOUBLE | GLUT_RGB);
glutInitWindowSize (300, 300);
glutInitWindowPosition (100, 100);
glutCreateWindow ("DDA Line Drawing Algorithm");
init ();
glutDisplayFunc(display);
glutMainLoop();
```

return 0; /* ISO C requires main to return int. */

}