```
#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 LineDDA(int X1, int Y1, int X2, int Y2)
{
  int dx=(X2-X1);
  int dy=(Y2-Y1);
  int steps;
  float xInc,yInc,x=X1,y=Y1;
  /* Find out whether to increment x or y */

  if(abs(dy)>abs(dx))
  {
    steps=abs(dy);
  }
  else
  {
    steps=abs(dx);
  }
  xInc=dx/(float)steps;
  yInc=dy/(float)steps;

  /* Plot the points */
  glBegin(GL_POINTS);
  /* Plot the first point */
  glVertex2d(x,y);
  int k;
  /* For every step, find an intermediate vertex */
  for(k=0;k<steps;k++)
  {
    x+=xInc;
    y+=yInc;
    glVertex2d(round_value(x), round_value(y));
  }
  glEnd();
```

```
  glFlush();
}
void display()
{
LineDDA(X1, Y1, X2, Y2);

}
void Init()
{
  /* Set clear color to white */
  glClearColor(1.0,1.0,1.0,0);
  /* Set fill color to black */
  glColor3f(0.0,0.0,0.0);
  /* glViewport(0 , 0 , 640 , 480); */
  /* glMatrixMode(GL_PROJECTION); */
  /* glLoadIdentity(); */
  /* Clears buffers to preset values */
  glClear(GL_COLOR_BUFFER_BIT);
  gluOrtho2D(0 , 640 , 0 , 480);
}
void main(int argc, char **argv)
{
  printf("Enter two end points of the line to be drawn:\n");
  printf("\n************************************");
  printf("\nEnter Point1( X1 , Y1):\n");
  scanf("%d%d",&X1,&Y1);
  printf("\n************************************");
  printf("\nEnter Point1( X2 , Y2):\n");
  scanf("%d%d",&X2,&Y2);

  /* Initialise GLUT library */
  glutInit(&argc,argv);
  /* Set the initial display mode */
  glutInitDisplayMode(GLUT_SINGLE | GLUT_RGB);
  /* Set the initial window position and size */
  glutInitWindowPosition(0,0);
  glutInitWindowSize(640,480);
  /* Create the window with title "DDA_Line" */
  glutCreateWindow("DDA_Line");
  /* Initialize drawing colors */
  Init();
  /* Call the displaying function */
  glutDisplayFunc(display);
  /* Keep displaying untill the program is closed */
  glutMainLoop();
}
```