Mapping of DDA Line Drawing Algorithm

DDA	DDA in OpenGL
Consider one point of the line as (X1,Y1) and the second point of the line as (X2,Y2). // calculate dx , dy **dx = X2 - X1;** **dy = Y2 - Y1;** **// Depending upon absolute value of dx & dy** **// choose number of steps to put pixel as** **// steps = abs(dx) > abs(dy) ? abs(dx) : abs(dy)** steps = abs(dx) > abs(dy) ? abs(dx) : abs(dy); // calculate increment in x & y for each steps Xinc = dx / (float) steps; Yinc = dy / (float) steps; // Put pixel for each step X = X1; Y = Y1; **putpixel (X,Y,WHITE);//plot the point** for (int i = 0; i <= steps; i++) { X += Xinc; Y += Yinc; putpixel (X,Y,RED); }	#include<GL/glut.h> #include<stdlib.h> #include<stdio.h> **void putpixel(float x, float y)** { **glBegin(GL_POINTS);** **glVertex2i(x,y);** **glEnd();** } float x1,x2,y1,y2; **void DDALine(void)** { float dy,dx,steps,x,y,k,Xinc,Yinc; **dx=x2-x1;** **dy=y2-y1;** **if(abs(dx)> abs(dy))** { steps = **abs**(dx); } **else** steps = **abs**(dy); Xinc = dx/steps; Yinc = dy/steps; x= x1; y=y1; **putpixel(x,y); //user defined function to plot point** **for** (k=0 ;k<steps;k++) { x= x + Xinc; y= y + Yinc; **putpixel(x,y);** }

```
glFlush();
}
```

```
void init(void)
{
glClearColor(0,0,0,0);
glColor3f(1.0,1.0,0.0);
gluOrtho2D(-100,100,-100,100);
}

int main(int argc, char** argv)
{
   printf("Enter the value of x1 : ");
   scanf("%f",&x1);
   printf("Enter the value of y1 : ");
   scanf("%f",&y1);
   printf("Enter the value of x2 : ");
   scanf("%f",&x2);
   printf("Enter the value of y2 : ");
   scanf("%f",&y2);

glutInit(&argc, argv);
glutInitDisplayMode (GLUT_SINGLE |
GLUT_RGB);

glutInitWindowSize (500, 500);
glutInitWindowPosition (100,100);

glutCreateWindow ("DDA Line Algo");

init();

glutDisplayFunc(DDALine);
glutMainLoop();

return 0;
}
```