DDA LINE:

```
#include<GL/glut.h>
#include<iostream>
#include<math.h>
using namespace std;
float r, g, b, x, y;
float x_1,x_2,y_1,y_2;
float xin, yin,length;
bool flag = true;
void mouse(int button, int state, int mousex, int mousey)
{
if(button GLUT_LEFT_BUTTON
&& state = GLUT_DOWN) {
flag = true;
x = mousex;
y = 640 - mousey;
}
}
int sgn(float a) {
if(a= 0) {
return 0;
}
if(a < 0) {
return -1;
}
else
```

```
return 1;
}
void Line() {
cout<<"x_1="<<x_1<<"y_1="<<y_1;
cout<<"x_2="<<x_2<<"y_2="<<y_2;
float dy, dx, length;
x_2 = x;
y_2 = y;
dy=y_2 - y_1;
dx = x_2 - x_1;
if(abs(dx)>=abs(dy)) {
length = abs(dx);
}
else {
length = abs(dy);
}
float xin, yin;
xin=(x_2-x_1)/length;
yin=(y_2y_1)/length;
float x, y;
x=x_1+0.5*sgn(xin);
y=y_1+0.5*sgn(yin);
int i=0;
while(i<=length)
{
glBegin(GL_POINTS);
```

```
glVertex2i(x,y);
glEnd();
x=x+xin;
y=y+yin;
i++;
}
glFlush();
}
void init(void) { glClearColor(0,0,0,0); gluOrtho2D(0,640,0,640);
glColor3f(1.0,1.0,0.0);
glClear(GL_COLOR_BUFFER_BIT);
}
int main(int argc, char** argv) { cout<<"Enter x1,y1 point";
cin>>x_1>>y_1;
glutInit(&argc,argv);
glutInitDisplayMode(GLUT_SINGLE| GLUT_RGB);
glutInitWindowSize(0,640);
glutCreateWindow("DDA LINE");
init();
glutMouseFunc(mouse);
glutDisplayFunc(Line);
glutMainLoop();
return 0;
}
```