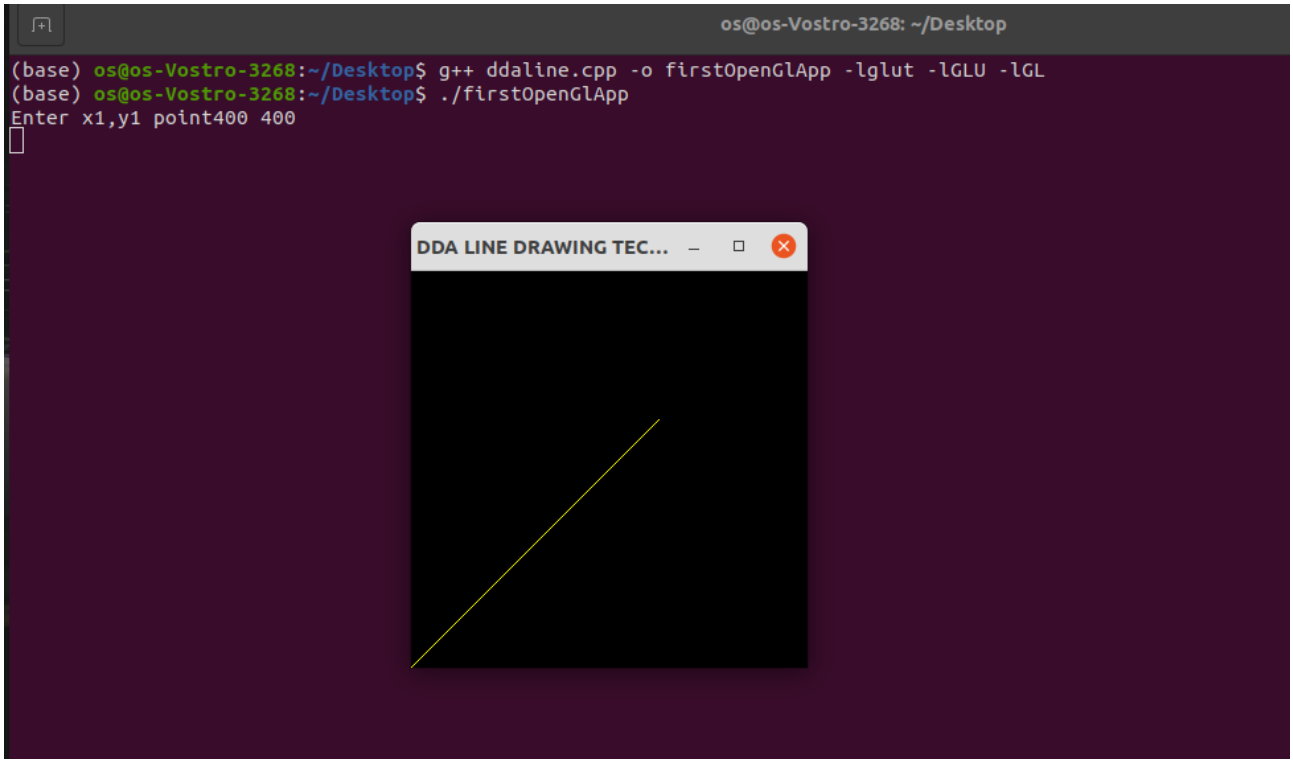
DDA LINE

```
#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_2 - y_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);
glColor3f(1.0,1.0,0.0);
gluOrtho2D(0,640,0,640);
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 DRAWING TECHNIQUE");
 init();
 glutMouseFunc(mouse);
 glutDisplayFunc(Line);
 glutMainLoop();
 return 0;
 }
```

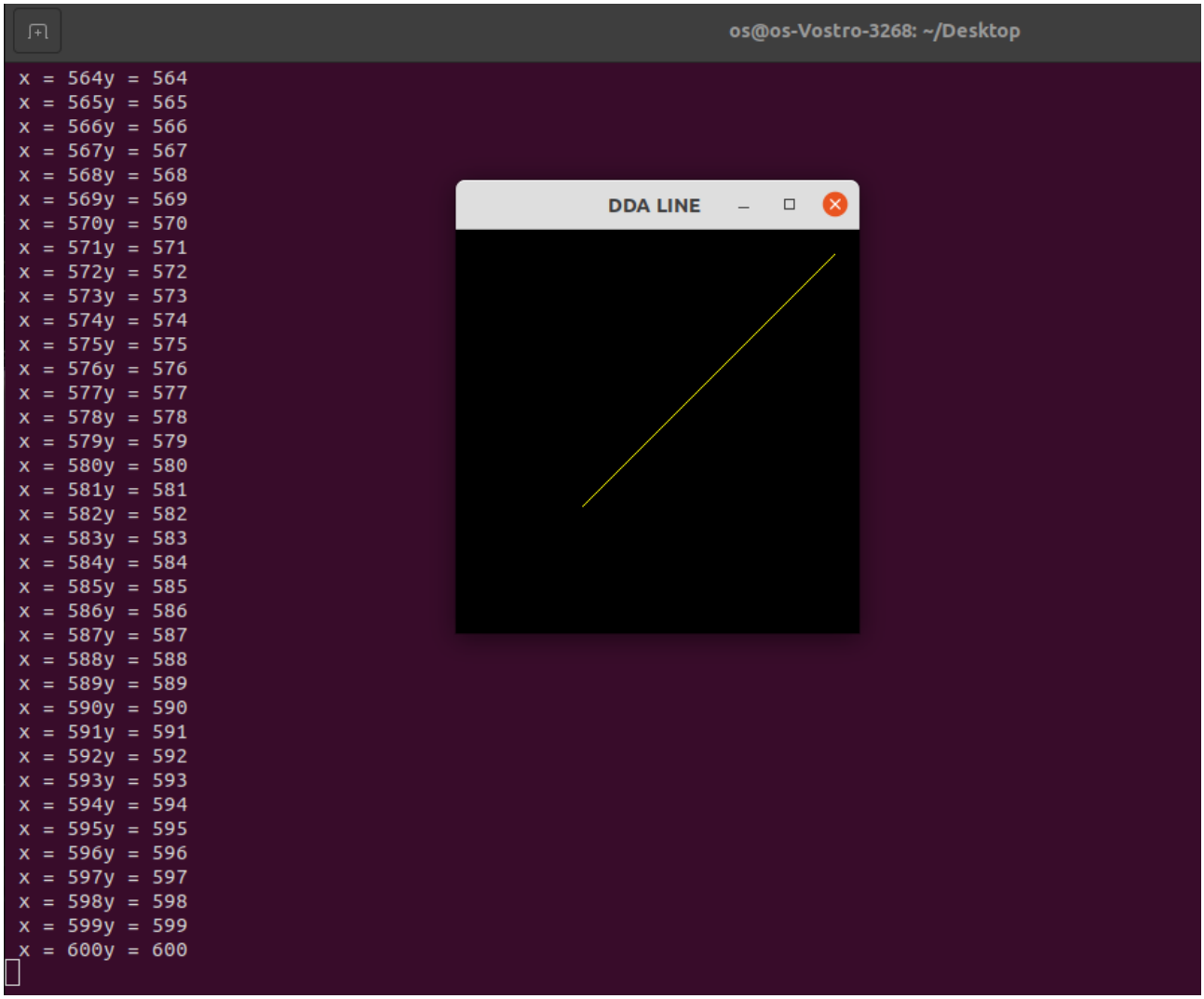


BRESENHAM LINE

```
#include<iostream>
#include<GL/glut.h>
#include<math.h>
using namespace std;
float r,g,b,x,y;
float x_1,x_2,y_1,y_2;

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 = 480-mousey;
        }
        cout<<"mousex = "<<x;
        cout<<"mousey = "<<y;
}

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 dx,dy,length,G;
	//x_2 = x;
	//y_2 = y;
	dy = y_2 - y_1;
	dx = x_2 - x_1;
	G = (2*dy)-dx;

	if(abs(dx) >= abs(dy)){
		length = abs(dx);
	}
	else{
	length = abs(dy);
	}
	int j =0;
	x = x_1;
	y = y_1;

	while(j <= length){
		if(abs(dx) >= abs(dy)){
			x = x+1;
			if(G>=0){
			y = y+1;
			G = G+2*(dy-dx);
			}
			else{
			G = G + (2*dy);
			}
		}
		else{
			y = y+1;
			if(G>=0){
			x = x+1;
			G = G+2*(dy-dx);
			}
			else{
			G = G+ (2*dy);
			}
		}
		cout<< "\n x = " << x;
		cout<< "y = " << y;
```

```
                glBegin(GL_POINTS);
                glVertex2i(x,y);
                glEnd();
                j++;
        }
        glFlush();

}

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

int main(int argc, char **argv)
{
        cout<<"Enter x1,y1 point";
 cin>>x_1>>y_1;
 cout<<"Enter x2,y2 point";
 cin>>x_2>>y_2;
 glutInit(&argc,argv);
 glutInitDisplayMode(GLUT_SINGLE| GLUT_RGB);
 glutInitWindowSize(0,600);
 glutCreateWindow("DDA LINE ");
 init();
 //glutMouseFunc(mouse);
 glutDisplayFunc(Line);
 glutMainLoop();
 return 0;
}
```


os@os-Vostro-3268: ~/Desktop
x = 564y = 564
x = 565y = 565
x = 566y = 566
x = 567y = 567
x = 568y = 568
x = 569y = 569
x = 570y = 570
x = 571y = 571
x = 572y = 572
x = 573y = 573
x = 574y = 574
x = 575y = 575
x = 576y = 576
x = 577y = 577
x = 578y = 578
x = 579y = 579
x = 580y = 580
x = 581y = 581
x = 582y = 582
x = 583y = 583
x = 584y = 584
x = 585y = 585
x = 586y = 586
x = 587y = 587
x = 588y = 588
x = 589y = 589
x = 590y = 590
x = 591y = 591
x = 592y = 592
x = 593y = 593
x = 594y = 594
x = 595y = 595
x = 596y = 596
x = 597y = 597
x = 598y = 598
x = 599y = 599
x = 600y = 600
DDA LINE