```
#include <GL/glut.h>
#include <math.h>

using namespace std;
int W = 900, H = 900;
int CA, CB, X1, Y1;



void Start() {
    glClear(GL_COLOR_BUFFER_BIT);
    glFlush();
}
void Display() {
    glClearColor(0, 0, 0, 0);
    gluOrtho2D(-(W / 2), (W / 2), -(H / 2), (H / 2));
}
void Pixel(int X, int Y) {
    glPointSize(5);//ADECUA EL TAMAÑO DEL PIXEL
    glBegin(GL_POINTS);
    glColor3f(1, 0, 0);
    glVertex2f(X, Y);
    glEnd();
    glFlush();
}
void Plano() {
    for (int i = -(W / 2); i <= (W / 2); i++) {
        Pixel(i, 0);
    }
    for (int i = -(H / 2); i <= (H / 2); i++) {
        Pixel(0, i);
    }
}
void circunferencia(int a, int b, int x1, int y1) {
    float dx, dy, r;
    dx = (x1 - a);
    dy = (y1 - b);
    r = round(sqrt((pow(dx, 2)) + (pow(dy, 2))));
    y1 = abs(y1);
    for (x1 = 0; x1 <= y1; x1++) {
        y1 = round(sqrt(pow(r, 2) - pow(x1, 2)));
        Pixel((x1 + a), (y1 + b));
        Pixel((y1 + a), (x1 + b));
        Pixel((-x1 + a), (y1 + b));
        Pixel((-y1 + a), (x1 + b));
        Pixel((-y1 + a), (-x1 + b));
        Pixel((-x1 + a), (-y1 + b));
        Pixel((y1 + a), (-x1 + b));
        Pixel((x1 + a), (-y1 + b));
    }
}
void Mouse(int B, int S, int X, int Y) {
    Plano();
    X = X - 900 / 2;
    Y = 900 / 2 - Y;
    if ((S == GLUT_DOWN) && (B == GLUT_LEFT_BUTTON)) {
        CA = X;
```

```
                CB = Y;
                Pixel(CA, CB);
        }
        if ((S == GLUT_DOWN) && (B == GLUT_RIGHT_BUTTON)) {
                X1 = X;
                Y1 = Y;
                Pixel(X1, Y1);
                circunferencia(CA, CB, X1, Y1);
        }
}
int main(int argc, char* argv[]) {
        glutInit(&argc, argv);
        glutInitDisplayMode(GLUT_RGBA | GLUT_SINGLE);
        glutInitWindowPosition(650, 50);
        glutInitWindowSize(W, H);
        glutCreateWindow("LINE");
        Display();
        glutDisplayFunc(Start);
        glutMouseFunc(Mouse);
        glutMainLoop();
}
```