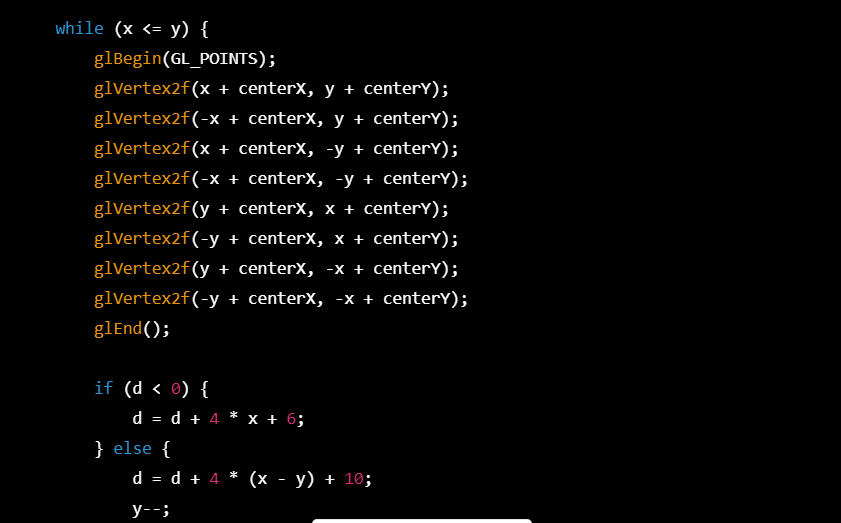
|  |  |
| --- | --- |
| Circunferencia de Bresenham | Del Valle Aragón Alexis rafael   * Computación gráfica e iteracción humano computadora * gruPo: 06 * fecha: 9/abril/2023   universidad nacional autónoma de méxico  facultad de ingeniería |

A través de ChatGPT realizar la búsqueda del algoritmo de una circunferencia de Bresenham y adaptarlo a un código para que sea funcional.





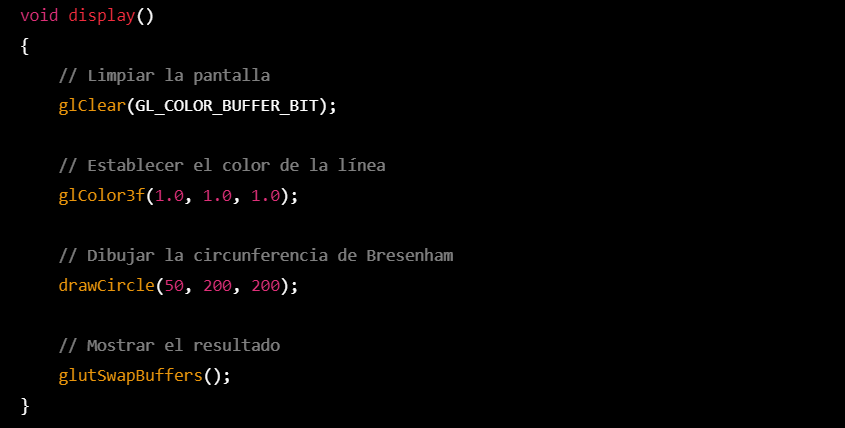




Ilustración Consulta de ChatGPT

Código de consulta:

#include <GL/glut.h>

#include <cmath>

// Función que dibuja la circunferencia utilizando el algoritmo de Bresenham

void drawCircle(int x0, int y0, int r) {

int x = 0;

int y = r;

int d = 3 - 2\*r;

glBegin(GL\_POINTS);

while (x <= y) {

glVertex2i(x0 + x, y0 + y);

glVertex2i(x0 + y, y0 + x);

glVertex2i(x0 - y, y0 + x);

glVertex2i(x0 - x, y0 + y);

glVertex2i(x0 - x, y0 - y);

glVertex2i(x0 - y, y0 - x);

glVertex2i(x0 + y, y0 - x);

glVertex2i(x0 + x, y0 - y);

if (d < 0) {

d = d + 4\*x + 6;

} else {

d = d + 4\*(x-y) + 10;

y--;

}

x++;

}

glEnd();

}

// Función que se encarga de renderizar la escena

void display() {

glClear(GL\_COLOR\_BUFFER\_BIT);

glColor3f(1.0, 1.0, 1.0); // Establecemos el color de la circunferencia a blanco

drawCircle(200, 200, 100); // Dibujamos la circunferencia centrada en (200, 200) con radio 100

glFlush();

}

// Función principal

int main(int argc, char\*\* argv) {

glutInit(&argc, argv);

glutCreateWindow("Circunferencia de Bresenham");

glutInitWindowSize(400, 400);

glutInitWindowPosition(100, 100);

glutDisplayFunc(display);

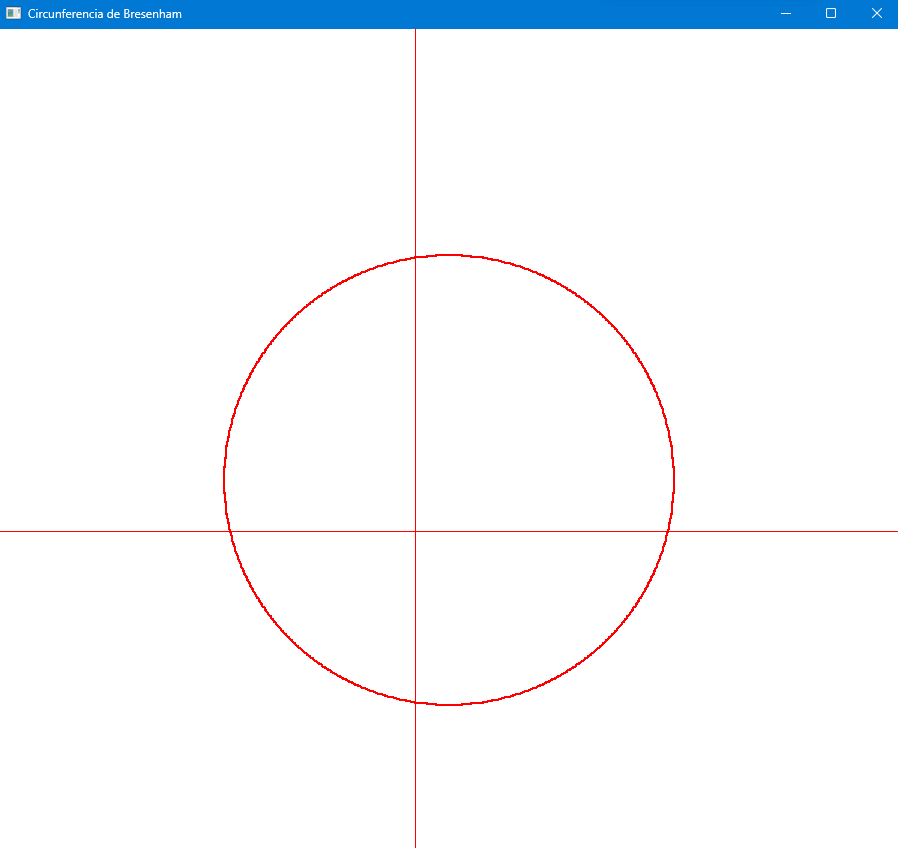
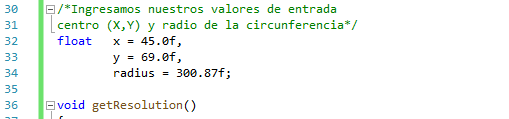
glutMainLoop();

return 0;

}

Dado que por si solo no funcionaba, el código brindado por ChatGPT no ofrece ninguna solución inicial.

Resultado final:



Circunferencia con centro (45,69) y radio= 300.87.