



CIRCUNFERENCIA DE BRESENHAM

DEL VALLE ARAGÓN ALEXIS RAFAEL

- COMPUTACIÓN GRÁFICA E
INTERACCIÓN 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.



```

while (x <= y) {
    glBegin(GL_POINTS);
    glVertex2f(x + centerX, y + centerY);
    glVertex2f(-x + centerX, y + centerY);
    glVertex2f(x + centerX, -y + centerY);
    glVertex2f(-x + centerX, -y + centerY);
    glVertex2f(y + centerX, x + centerY);
    glVertex2f(-y + centerX, x + centerY);
    glVertex2f(y + centerX, -x + centerY);
    glVertex2f(-y + centerX, -x + centerY);
    glEnd();

    if (d < 0) {
        d = d + 4 * x + 6;
    } else {
        d = d + 4 * (x - y) + 10;
        y--;
    }
}

```

```

void display()
{
    // Limpiar la pantalla
    glClear(GL_COLOR_BUFFER_BIT);

    // Establecer el color de la línea
    glColor3f(1.0, 1.0, 1.0);

    // Dibujar la circunferencia de Bresenham
    drawCircle(50, 200, 200);

    // Mostrar el resultado
    glutSwapBuffers();
}

```

```

int main(int argc, char **argv)
{
    // Inicializar la ventana
    glutInit(&argc, argv);
    glutInitDisplayMode(GLUT_DOUBLE | GLUT_RGB);
    glutInitWindowSize(400, 400);
    glutCreateWindow("Circunferencia de Bresenham");

    // Establecer la función de dibujo
    glutDisplayFunc(display);

    // Iniciar el bucle principal
    glutMainLoop();

    return 0;
}

```

Ilustración 1 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();
}

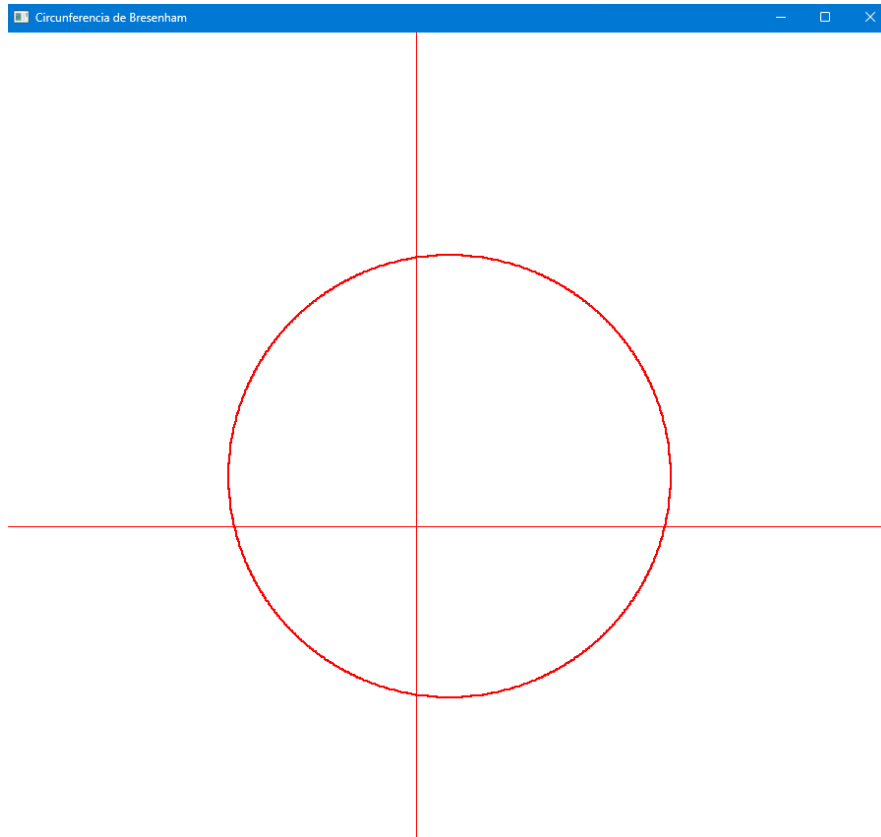
// Función principal
int main(int argc, char** argv) {
    glutInit(&argc, argv);
    glutCreateWindow("Circunferencia de Bresenham");
    glutInitWindowSize(400, 400);
    glutInitWindowPosition(100, 100);
    glutDisplayFunc(display);
    glutMainLoop();
    return 0;
}

```

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

Resultado final:

```
30  /*Ingresamos nuestros valores de entrada
31  | centro (X,Y) y radio de la circunferencia*/
32  float   x = 45.0f,
33          y = 69.0f,
34          radius = 300.87f;
35
36  void getResolution()
37  {
```



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