Source Code

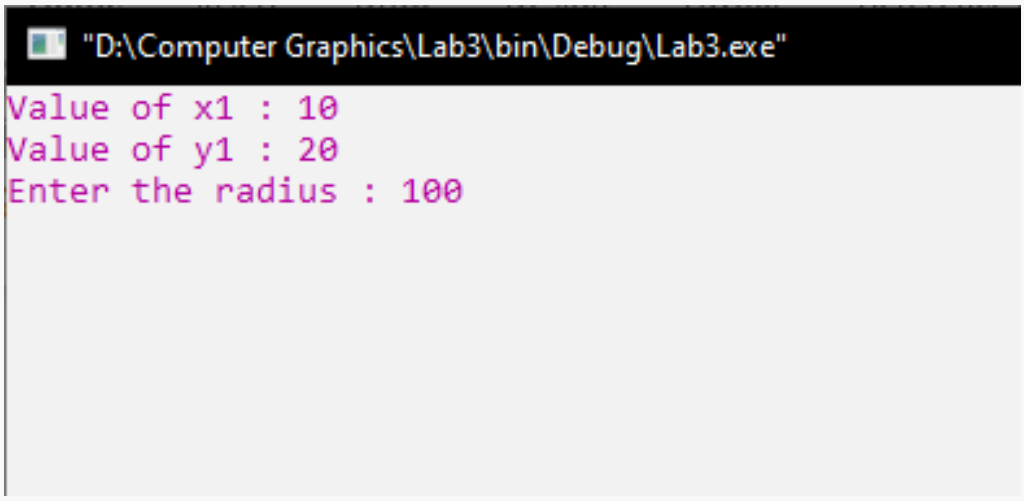
```
#include<stdlib.h>

#include<stdio.h>

#include <GL/gl.h>

#include <GL/glut.h>


float xc, yc, r;


void TakeInput()

{

printf("Value of x1 : ");

scanf("%f", & xc);

printf("Value of y1 : ");

scanf("%f", & yc);

printf("Enter the radius : ");

scanf("%f", & r);

}


void display(void)

{

float p = 1 - r;

float x = 0;
```

```
float y = r;
glBegin(GL_POINTS);
glVertex2f(xc + x, yc + y);
glVertex2f(xc + y, yc + x);
glVertex2f(xc + x, yc - y);
glVertex2f(xc - y, yc - x);
while(x <= y)
{
if(p < 0)
{
x = x + 1;
p = p + 2.0 * x + 1;
}
else
{
x = x + 1;
y = y - 1;
p = p + 2 * (x - y) + 1;
}
glVertex2f(xc + x, yc + y);
glVertex2f(xc + x, yc - y);
glVertex2f(xc - x, yc + y);
```

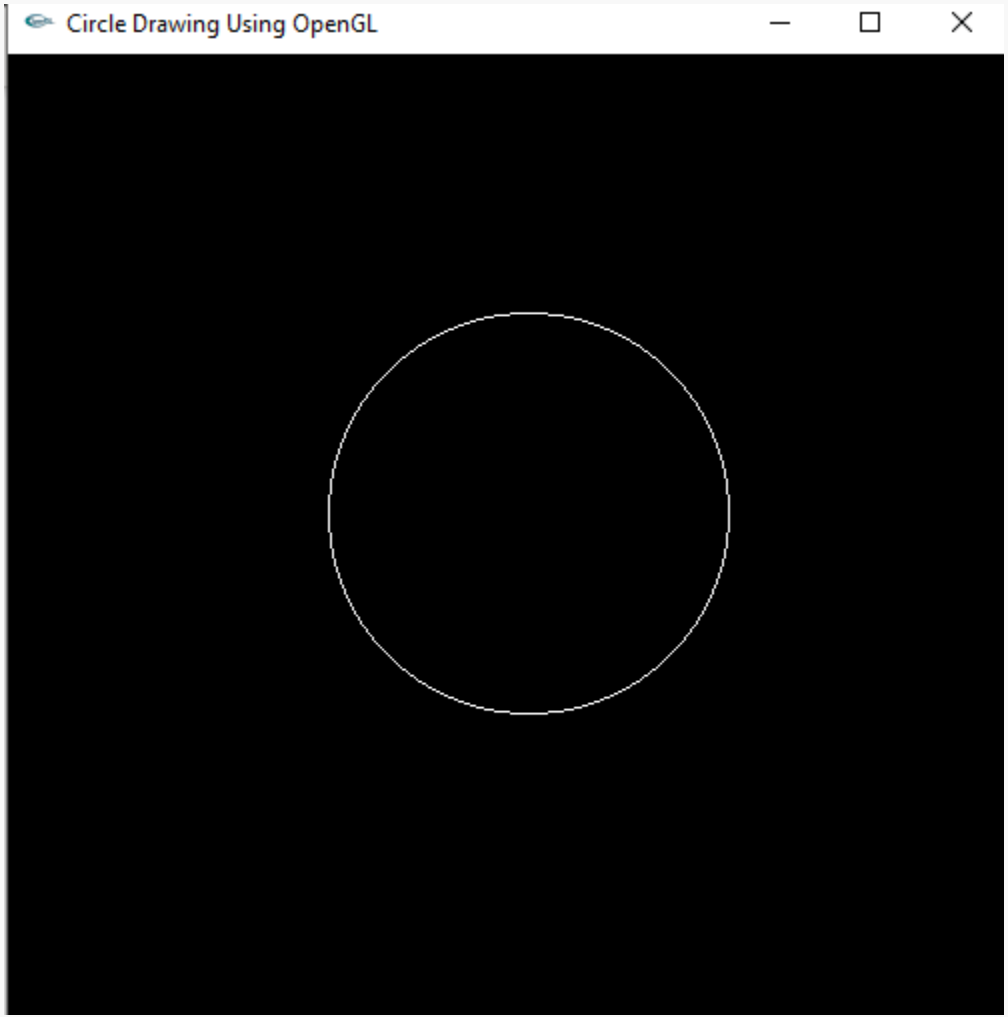
```
glVertex2f(xc - x, yc - y);

glVertex2f(xc + y, yc + x);

glVertex2f(xc - y, yc + x);

glVertex2f(xc + y, yc - x);

glVertex2f(xc - y, yc - x);

}

glEnd();

glFlush();

}

int main(int argc, char ** argv)

{

TakeInput();


glutInit(&argc, argv);

glutInitDisplayMode(GLUT_SINGLE | GLUT_RGB);

glutInitWindowSize(500, 500);

glutInitWindowPosition(100, 150);

glutCreateWindow("Circle Drawing Using OpenGL");


glClearColor(0, 0, 0, 0);

glClear(GL_COLOR_BUFFER_BIT);

gluOrtho2D(-250, 250, -250, 250);
```

```
glMatrixMode(GL_PROJECTION);

glViewport(0, 0, 500, 500);



glutDisplayFunc(display);

glutMainLoop();

return 0;

}
```

Input:



Output:



Chittagong University of Engineering and Technology

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

COURSE CODE: CSE - 458

COURSE TITLE: COMPUTER GRAPHICS (SESSIONAL)

EXPERIMENT NO: 04

NAME OF THE EXPERIMENT: Implementation of Mid-Point Circle Drawing Algorithm in OpenGL.

DATE OF SUBMISSION: 07 June 2023

REMARKS

Submitted By:

STUDENT NAME : SALMAN FARSI

STUDENT ID : 1804102

GROUP : B2