

CSO351: Computer Graphics

Lab Assignment 2.Ellipse.(a): Ellispe Generation using Parametric Equation

Objective:

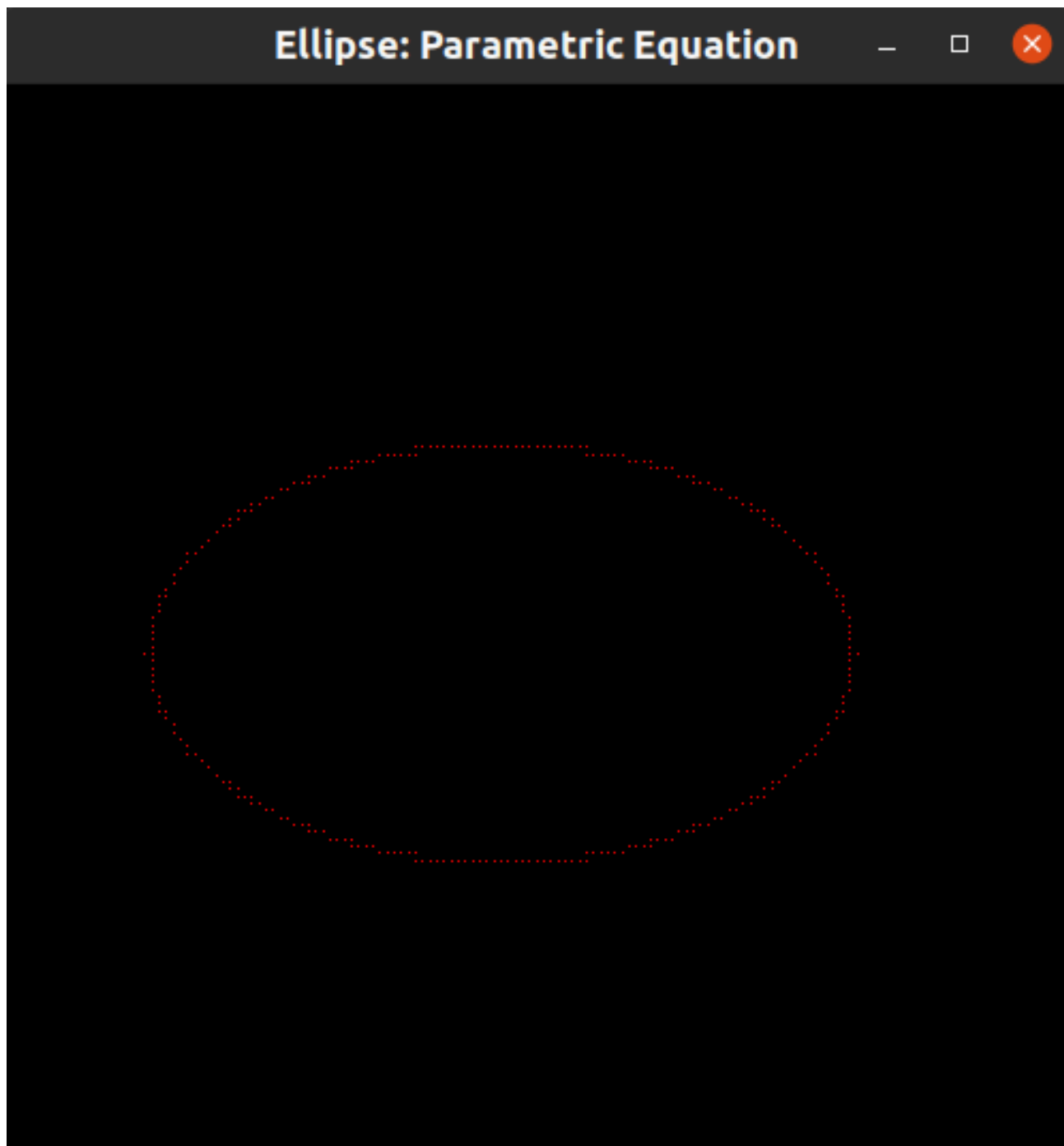
Write a program in C/C++ for implementation of ellipse generation using parametric equation.

Algorithm:

- **Step 1:** Take input the coordinates of the origin, the lengths of major and minor axes for the ellipse.
- **Step 2:** Set initial coordinates as $(a, 0)$ and plot the point. Set angle θ as 0.
- **Step 3:** Calculate the x and y coordinates of the next point by using the parametric equation of the ellipse $x = a \cos(\theta)$ and $y = b \sin(\theta)$.
- **Step 4:** Plot the point (x, y) , $(x, -y)$, $(-x, y)$ and $(-x, -y)$. Increment θ by 1.
- **Step 5:** Repeat step 3 and step 4 while the θ is less than or equal to 90 degrees.

Result:**Input:**

```
swaraj@shiv-raj-75:~/Documents/Assignments/Sem5/CG$ ./2.ellipse.a  
Enter the center: 20 20  
Enter major and minor axes: 50 30
```

Output:

Conclusion:

- This concludes that the algorithm is very slow because the trigonometric functions are slow.
- Ellipse is not at all smooth as seen in the generated plot.

Appendix: Code

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

using namespace std;
int pntX1, pntY1, a, b;

void plot(int x, int y)
{
    glBegin(GL_POINTS);
    glVertex2i(x + pntX1, y + pntY1);
    glEnd();
}

void init(void)
{
    glClearColor(0.0, 0.0, 0.0, 0.0);
    glMatrixMode(GL_PROJECTION);
    glLoadIdentity();
    gluOrtho2D(-50, 100, -50, 100);
}

void midPointCircleAlgo()
{
    double x = a;
    double y = 0;
    int theta = 0;
    while (theta <= 90)
    {
        plot(x, y);
        plot(-x, y);
        plot(x, -y);
        plot(-x, -y);
        x = a * cos(theta * 3.14 / 180.0);
        y = b * sin(theta * 3.14 / 180.0);
        theta++;
    }
}

void display(void)
{
    glClear(GL_COLOR_BUFFER_BIT);
```

```
    glColor3f(1.0, 0.0, 0.0);
    glPointSize(1.0);
    midPointCircleAlgo();
    glFlush();
}

int main(int argc, char **argv)
{
    cout << "Enter the center: ";
    cin >> pntX1 >> pntY1;
    cout << "Enter major and minor axes: ";
    cin >> a >> b;
    glutInit(&argc, argv);
    glutInitDisplayMode(GLUT_SINGLE | GLUT_RGB);
    glutInitWindowSize(500, 500);
    glutInitWindowPosition(0, 0);
    glutCreateWindow("Ellipse: Parametric Equation");
    glutDisplayFunc(display);
    init();
    glutMainLoop();
    return 0;
}
```