Gokul Krishna R Program 4 26/10/2022
20220040

Bresenham’s Algorithm

Code:

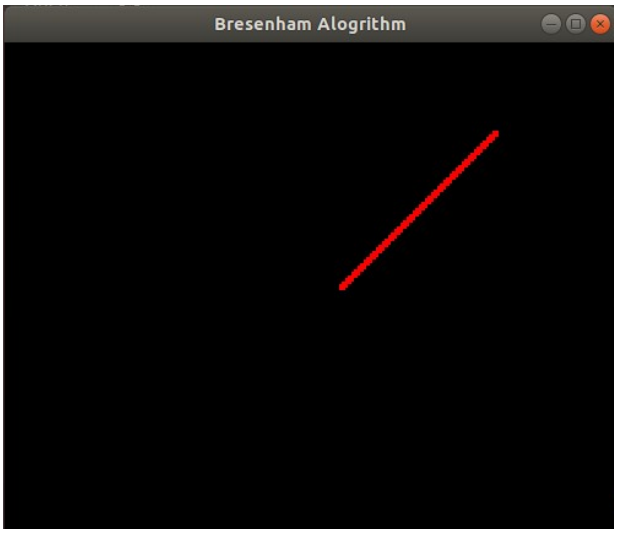
```
from OpenGL.GL import *
from OpenGL.GLUT import *
from OpenGL.GLU import *
import sys

WINDOW_SIZE=500
SCALE=100
x1=y1=0
x2=y2=25
def bresenham():
    glClear(GL_COLOR_BUFFER_BIT)
    glColor3f(1.0,0.0,0.0)
    glPointSize(5)
    glBegin(GL_POINTS)
    global x1,x2,y1,y2
    glVertex2f(x1/SCALE,y1/SCALE)
    dx,dy=abs(x2-x1),abs(y2-y1)

    if(dx>dy):
        p=(2*dy)-dx
        y=y1
        for x in range (x1+1,x2+1):
            if(p<0):
                p+=2*dy
            else:
                p+=(2*dy)-(2*dx)
                y+=1
            glVertex2f(x/SCALE,y/SCALE)
    else:
        p=(2*dx)-dy
        x=x1
        for y in range (y1+1,y2+1):
            if(p<0):
                p+=2*dx
            else:
                p+=(2*dx)-(2*dy)
                x+=1
            glVertex2f(x/SCALE,y/SCALE)
    glEnd()
    glFlush()
def main():
    glutInit(sys.argv)
    glutInitDisplayMode(GLUT_SINGLE,GLUT_RGB)
    glutInitWindowSize(WINDOW_SIZE,WINDOW_SIZE)
    glutInitWindowPosition(50,50)
    global x1,x2,y2,y1
    x1=int(input("Enter x1: "))
    y1=int(input("Enter y1: "))
    x2=int(input("Enter x2: "))
    y2=int(input("Enter y2: "))
    if(x1>x2):
        x1,x2=x2,x1
        y1.y2=y2,y1
    glutCreateWindow("Bresenham Alogrithm")
    glutDisplayFunc(bresenham)
    glutMainLoop()
main()
```

Output:

```
Enter x1: 10
Enter y1: 20
Enter x2: 60
Enter y2: 70
```


Bresenham Alogrithm