

NAMA : Christopher Juliano Wiweka Sihombing

KELAS : 2020B

NIM : 20051397050

1. Source Code

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

def init():
    glClearColor(0.0,0.0,0.0,1.0)
    gluOrtho2D(0,20,0,20)
    glColor3f(128.0,0.0, 0.0)
    glPointSize(10.0)
    glMatrixMode(GL_PROJECTION)
    glLoadIdentity()

def bresenham(x1,y1,x2,y2):
    #hitung dx dan dy
    dx = abs(x2 - x1)
    dy = abs(y2 - y1)

    #hitung p
    p = 2 * dy - dx
    duady = 2 * dy
    duadydx = 2 * (dy - dx)

    #tentukan titik awal dan akhir
    if (x1 > x2):
        x = x2
        y = y2
        xend = x1
    else:
        x = x1
        y = y1
        xend = x2

    #gambar titik awal
    glBegin(GL_POINTS)
    glVertex2i(x, y)

    #perulangan untuk menggambar titik-titik
```

```

while (x < xend):
    x = x+1
    if (p < 0):
        p += duady
    else:
        if (y1 > y2):
            y = y-1
        else:
            y = y+1
        p += duadydx
    glVertex2i(x, y)

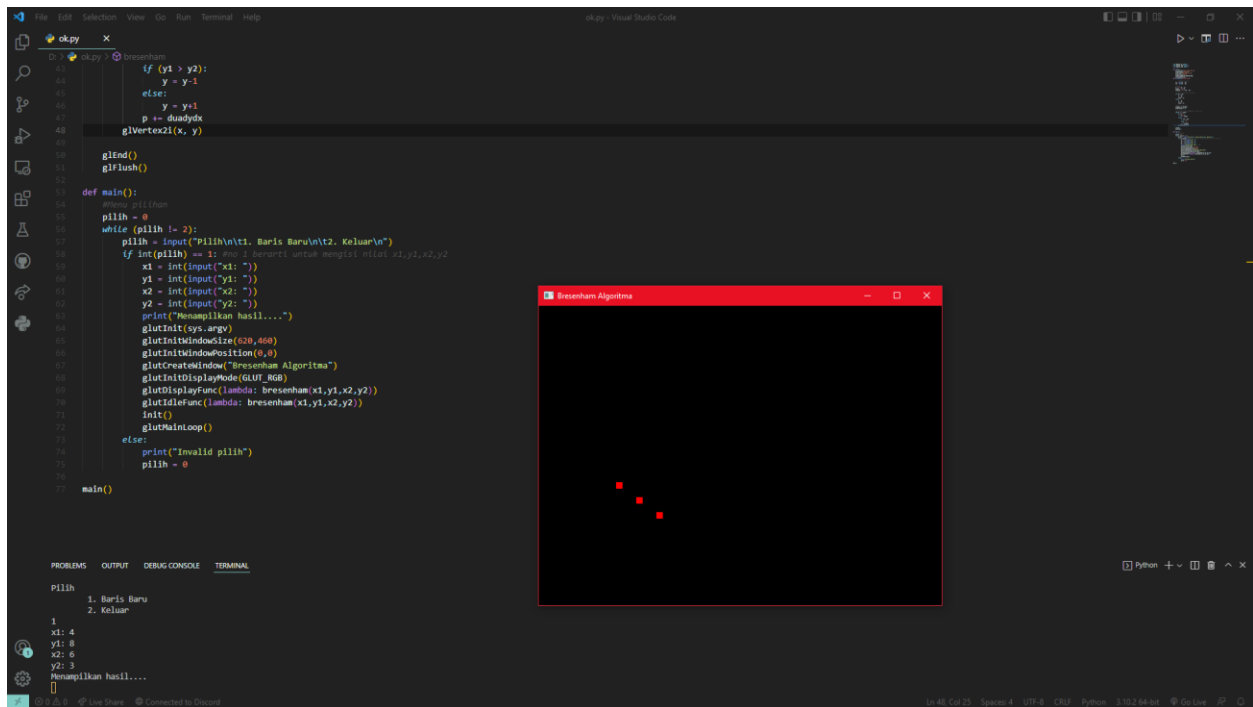
glEnd()
glFlush()

def main():
    #Menu pilihan
    pilih = 0
    while (pilih != 2):
        pilih = input("Pilih\n\t1. Baris Baru\n\t2. Keluar\n")
        if int(pilih) == 1: #no 1 berarti untuk mengisi nilai x1,y1,x2,y2
            x1 = int(input("x1: "))
            y1 = int(input("y1: "))
            x2 = int(input("x2: "))
            y2 = int(input("y2: "))
            print("Menampilkan hasil....")
            glutInit(sys.argv)
            glutInitWindowSize(620,460)
            glutInitWindowPosition(0,0)
            glutCreateWindow("Bresenham Algoritma")
            glutInitDisplayMode(GLUT_RGB)
            glutDisplayFunc(lambda: bresenham(x1,y1,x2,y2))
            glutIdleFunc(lambda: bresenham(x1,y1,x2,y2))
            init()
            glutMainLoop()
        else:
            print("Invalid pilih")
            pilih = 0

main()

```

2. Hasil output



The screenshot shows a Visual Studio Code editor with a Python file named `ok.py`. The code implements Bresenham's algorithm for drawing a line. It includes a `bresenham` function that takes `x1, y1, x2, y2` as input and returns a list of points. The `main` function prompts the user to enter two points and then calls `bresenham` to draw the line. The output window shows the following text:

```
Pilih
1. Baris Baru
2. Keluar
1
x1: 4
y1: 8
x2: 6
y2: 3
Menampilkan hasil....
```

The output window also shows a visual representation of the line drawn by the algorithm, with red dots representing the points along the line from (4, 8) to (6, 3).

3. Cara kerja algoritma bresenham

membuat baris dari 2 titik koordinat dengan cara membuat titik di sepanjang jalur dari titik awal menuju titik akhir.

Urutan algoritma

1. Menentukan titik awal (x_1, y_1) (x_2, y_2)
2. Menghitung dx , dy dan $2(dx-dy)$
3. Menentukan titik awal ($x_1 > x_2$) $x=x_2$ $y=y_2$ $xend=x_1$ atau $x=x_1$ $y=y_1$ $xend=x_2$
4. Menggambar titik dengan ($x < xend$) dengan cara $x=x+1$ jika $p < 0$ $p+=2dy$
Kemudian ($y_1 > y_2$) $y=y-1$ atau $y=y+2$ $p+=2dydx$