

Laporan Bresenham
Mata Kuliah Grafika Komputer



Dibuat oleh :

Gagah Rizky Mulyawan (20051397045)

2020A

Program Studi D4 Manajemen Informatika

Fakultas Vokasi

Universitas Negeri Surabaya

2022

Algoritma Bresenham

Algoritma Garis Bresenham adalah suatu garis yang menentukan titik-titik dalam dimensi yang membentuk dekat dengan garis lurus antara dua titik yang diberikan. Pendekatan yang biasa digunakan untuk menggambar garis pada layar komputer, karena hanya menggunakan penambahan bilangan bulat, pengurangan dan pergeseran. Algoritma ini merupakan salah satu algoritma paling awal yang dikembangkan di bidang komputer grafis.

Cara kerja Algoritma Bresenham adalah mengecek garis yang sudah diubah hanya dengan menggunakan metode perhitungan integer yang nantinya akan terus bertambah sehingga bisa menampilkan bentuk lingkaran dan bentuk kurva lainnya.

Langkah-langkah :

1. Masukkan 2 titik, kemudian simpan titik yang paling kiri sebagai nilai (X,Y)
2. Plotkan titik yang pertama.
3. Hitung Δx , Δy , $2\Delta y$ dan $2\Delta y - 2\Delta x$ dan dapatkan nilai awal parameter keputusan sbb:
$$p_0 = 2\Delta y - \Delta x$$
4. Setiap X_k sepanjang garis, mulai dari $k=0$, lakukan langkah pengujian sbb : jika $p_k < 0$, maka titik selanjutnya yang akan diplot adalah (X_{k+1}, Y_k) , kemudian :
$$p_{k+1} = p_k + 2\Delta y$$
 jika sebaliknya, maka titik selanjutnya memiliki nilai (X_{k+1}, Y_{k+1}) , lalu perhitungannya : $p_{k+1} = p_k + 2\Delta y - \Delta x$
5. Ulangi langkah ke-4 sebanyak Δx kali.

Source Code

```

# Gagah Rizky Mulyawan
# 20051397045
# D4 MI 2020A

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

✓ def BRESENHAM(x1,y1,x2,y2):
    #menentukan delta X dan delta Y
    x=x1
    y=y1
    deltaX = abs(x2-x1)
    deltaY = abs(y2-y1)

    #menghitung p, 2dx dan 2(dy-dx)
    p = (2 * deltaY) - (deltaX)
    duadx = 2 * deltaX
    duaDyDx = 2 * (deltaY-deltaX)

    #Menentukan titik awal dan titik akhir
    ✓ if(x1>x2):
        x = x2
        y = y2
        xend = x1

    ✓ else:
        x = x1
        y = y1
        xend = x2

```

```

#Memulai menggambar menggunakan BRESENHAM
#Membersihkan window
glClear(GL_COLOR_BUFFER_BIT)
#Menentukan warna
glColor3f(1.0,0.0,9.0)
#Spesifikasikan diameter dari pixel yang akan digambar
glPointSize(9.0)
#Memilih mode point
glBegin(GL_POINTS)

#Looping pada saat nilai x1 < x2
while x<xend:
    #Menentukan titik yang akan diisi
    x +=1
    S
    if(p<0):
        p+=duadx
    elif(y1>y2):
        y-=1
    else:
        y+=1

    p += duaDyDx

    #Menggambar pixel
    glVertex2i(x,y)

glEnd()
glFlush()

```

```

def main():
    x1 = int(15)
    y1 = int(10)
    x2 = int(50)
    y2 = int(35)

    #inisialisasi glut
    glutInit(sys.argv)
    #inisialisasi tipe display glut
    glutInitDisplayMode(GLUT_RGB)
    #inisialisasi ukuran layar glut
    glutInitWindowSize(700,700)
    #inisialisasi posisi layar glut
    glutInitWindowPosition(0,0)
    #inisialisasi pembuatan window
    glutCreateWindow("Menggambar garis menggunakan BRESENHAM")
    glutDisplayFunc(lambda: BRESENHAM(x1,y1,x2,y2))
    glutIdleFunc(lambda: BRESENHAM(x1,y1,x2,y2))

    #Memberikan layar dan memberikan warna
    glClearColor(0.0,0.0,0.0,1.0)
    #Set origin dari grid dan ukurannya 100 x 100
    gluOrtho2D(0,90,0,90)
    glutMainLoop()

main()

```

Output

