

**LAPORAN TUGAS PRATIUM**  
**MATA KULIAH GRAFIKA KOMPUTER**  
**MEMBUAT LINGKARAN MENGGUNAKAN ALGORITMA**  
**BRESENHAM**



Disusun Oleh :

Dilla Safira

20051397072

D4 MI 2020B

**PROGRAM STUDI MANAJEMEN INFORMATIKA**

**PROGRAM VOKASI**

**UNIVERSITAS NEGERI SURABAYA**

**2022**

## **A. PENGERTIAN ALGORITMA BRESENHAM**

Algoritma bresenham merupakan suatu algoritma (pendekatan) yang dikreasikan oleh bresenham yang tidak kalah akurat dan efisien dengan algoritma primitif lainnya (seperti DDA). Bagian pengkonversian (scan-knversi) garis akan melakukan kalkulasi untuk penambahan nilai-nilai integer (yang dibutuhkan untuk membentuk garis) yang disesuaikan dengan tipe grafik yang dipakai oleh layar komputer (keadaan monitor pc) kita. Untuk mengilustrasikan pendekatan bresenham, pertama kita harus memperhatikan proses scan- konvensi untuk garis dengan slope positif yang lebih kecil dari 1. Posisi pixel sepanjang line-path kemudian ditentukan dengan penyamplingan pada unit interval x. dimulai dari endpoint kiri ( $X_0, Y_0$ ) dari garis yang diberikan, kita pindahkan beberapa kolom berturut-turut (berdasarkan posisi x) dan plot pixel-pixel yang mempunyai nilai scan-line y ke jarak yang paling dekat dengan line-path.

## **B. ATURAN BRESENHAM**

Jika  $P_k$  bernilai positif (+), maka tambahkan hasilnya dengan B dan nilai x dan y ditambah 1.

Jika  $P_k$  bernilai negatif (-), maka tambahkan hasilnya dengan A dan nilai x ditambah 1, sedangkan y ditambah 0 (tetap).

Putaran dihentikan jika koordinat x dan y sudah mencapai batas akhir.

## **C. PRINSIP ALGORITMA BRESENHAM**

1. Sumbu vertikal memperlihatkan posisi scan line.
2. Sumbu horizontal memperlihatkan kolom pixel
3. Pada tiap langkah, penentuan pixel selanjutnya didasari oleh parameter integer yang nilainya proporsional dengan pengurangan antara vertical separations dari dua posisi piksel dari nilai actual.

## ❖ KODINGAN

```
Kodingan_Bresenham.py
1  #Nama          : Dilla Safira
2  #NIM           : 20051397072
3  #Kelas/Prodi  : 2020B/Manajemen Informatika
4
5  from OpenGL.GL import *
6  from OpenGL.GLU import *
7  from OpenGL.GLUT import*
8
9  def init():
10     glClearColor(0.0, 0.0, 0.0, 0.0)
11     gluOrtho2D(-50.0, 50.0, -50.0, 50.0)
12     glPointSize(5)
13
14  def plot(x, y):
15     glBegin(GL_POINTS)
16     glVertex2f(x, y)
17     glEnd()
18
19  def bresenham_drawing_circle(r):
20     # contoh soal jika nilai center circle (3,3) dan r = 9
21     x_center = 3
22     y_center = 3
23
24     r = 9
25     x = 0
26     y = r
27     # parameter keputusan bresenham
28     d = 3 - 2 * r
29     # membuat titik pada koordinat yang ditentukan
30     plot(x + x_center, y + y_center)
31     while y > x:
32         if d < 0:
33             x += 1
34             d += 4 * x + 6
35         else:
```

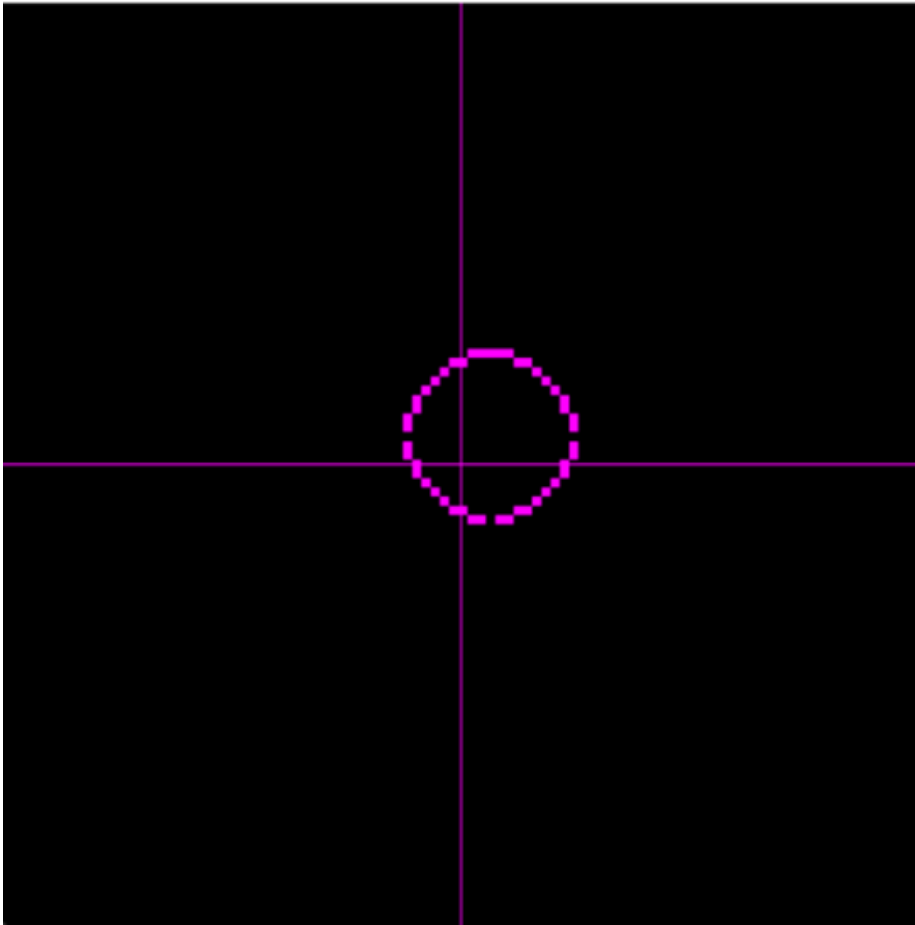
Kodingan\_Bresenham.py

```
34         d += 4 * x + 6
35     else:
36         x += 1
37         y -= 1
38         d += (4 * (x - y)) + 10
39     # Jika x <= y sudah terpenuhi, maka perulangan akan dihentikan.
40     # jika nilai (x,y)sudah didapatkan, lalu membalikan nilai (y, x)
41
42     # Untuk pixel (x, y)
43
44     # Quadrant 1
45     plot(x + x_center, y + y_center)
46
47     # Quadrant 2
48     plot(x + x_center, -y + y_center)
49
50     # Quadrant 3
51     plot(-x + x_center, -y + y_center)
52     # Quadrant 4
53
54     plot(-x + x_center, y + y_center)
55     # Untuk pixel (y, x)
56
57     # Quadrant 1
58     plot(y + x_center, x + y_center)
59
60     # Quadrant 2
61     plot(-y + x_center, x + y_center)
62
63     # Quadrant 3
64     plot(-y + x_center, -x + y_center)
65
66     # Quadrant 4
67     plot(y + x_center, -x + y_center)
68
69 def plotpoints():
70
71     glClear(GL_COLOR_BUFFER_BIT)
```

Kodingan\_Bresenham.py

```
67     plot(y + x_center, -x + y_center)
68
69 def plotpoints():
70
71     glClear(GL_COLOR_BUFFER_BIT)
72     glColor3f(1.0, 0.0, 1.0)
73
74     glBegin(GL_LINES)
75
76     glVertex2f(-50, 0)
77     glVertex2f(50, 0)
78
79     glVertex2f(0, -50)
80     glVertex2f(0, 50)
81
82     glEnd()
83
84     bresenham_drawing_circle(40)
85
86     glFlush()
87 def main():
88     glutInit(sys.argv)
89     glutInitDisplayMode(GLUT_SINGLE | GLUT_RGB)
90     glutInitWindowSize(500, 500)
91     glutInitWindowPosition(100, 100)
92     glutCreateWindow("Bresenham Circle - Dilla Safira")
93     glutDisplayFunc(plotpoints)
94
95     init()
96     glutMainLoop()
97 main()
98
```

## ❖ OUTPUT



## D. PENJELASAN KODINGAN

- OpenGL merupakan library yang digunakan untuk melakukan pemrograman grafik
- `glClearColor(0.0, 0.0, 0.0, 0.0)` fungsinya untuk mengedit warna background menjadi warna hitam
- `gluOrtho2D(-50.0, 50.0, -50.0, 50.0)` fungsinya untuk mengatur volume tampilan pada windows
- `glPointSize(5)` digunakan untuk mengatur besar titik yang akan digambar
- `def plot(x, y):` dimana `x` dan `y`: merupakan titik koordinat plot berupa variabel dengan panjang atau jumlah observasi yang sama.
- `def bresenham_drawing_circle(r):` digunakan untuk memberitahu bahwa grafik yang akan digambar adalah bentuk lingkaran. Diketahui  $r = 9$ ,  $x\_center = 3$ ,  $y\_center = 3$  dan dimana  $d = 3 - 2 * r$ .
- Quadrant 1 `plot(x + x_center, y + y_center)` memiliki penjelasan dimana

Quadrant berarti diagram center memiliki makna gambar lingkaran.

- `glVertex2f(-50, 0)` yaitu suatu fungsi untuk menggambar titik pada grafik lingkaran