

Algoritma Bresenham

Grafika Komputer



Disusun Oleh :

Dewi Nur Arifah – 20051397079

Manajemen Informatika 2020A

Algoritma Bresenham:

Algoritma bresenham merupakan suatu algoritma (pendekatan) yang dikreasikan oleh bresenham yang tidak kalah akurat dan efisien dengan algoritma primitif lainnya (seperti DDA).

Cara kerja dari algoritma ini adalah memeriksa garis yang telah diubah hanya dengan menggunakan perhitungan integer yang terus bertambah yang bisa diadaptasikan untuk menampilkan lingkaran dan bentuk kurva yang lain.

Langkah-langkah dalam algoritma Bresenham adalah:

1. Input dua titik, dan simpan titik yang paling kiri sebagai (x_0, y_0)
2. Plotkan titik pertama tersebut
3. Hitunglah Δx , Δy , $2\Delta y$ dan $2\Delta y - 2\Delta x$ serta perolehlah nilai awal parameter keputusan sbb: $p_0 = 2\Delta y - \Delta x$
4. Setiap x_k sepanjang garis, mulai dari $k = 0$, lakukan pengujian sbb:

Apabila $p_k < 0$, maka titik berikutnya yang akan diplot adalah (x_{k+1}, y_k) , kemudian : $p_{k+1} = p_k + 2\Delta y$
5. Apabila Sebaliknya, maka titik berikutnya bernilai (x_{k+1}, y_{k+1}) , lalu perhitungannya: $p_{k+1} = p_k + 2\Delta y - 2\Delta x$
6. Ulangi langkah 4 sebanyak Δx kali.

Source Code:

Algo_Bresenham.py 9+ bresenham.py 9+ bresenham dewi.py 9+ ×

C: > Users > user > Downloads > bresenham dewi.py > ...

```
1  # Algoritma Garis Bresenham
2  # Dewi Nur Arifah
3  # 20051397079 - 2020A D4 Manajemen Informatika
4
5  from OpenGL.GL import *
6  from OpenGL.GLU import *
7  from OpenGL.GLUT import *
8
9
10 def init():
11     glClearColor(0.0, 0.0, 0.0, 0.0)
12     gluOrtho2D(-100.0, 100.0, -100.0, 100.0)
13     glPointSize(10)
14
15
16 def plot(x, y):
17     glBegin(GL_POINTS)
18     glVertex2f(x, y)
19     glEnd()
20
21
22 def bresenham_circle_drawing(r):
23
24     # tempat lingkaran hasil output
25     x_position = 8
26     y_position = -8
27
28     x = 0
29     y = r
30
31     # parameter keputusan
32     d = 3 - 2 * r
```

C: > Users > user > Downloads > bresenham dewi.py > ...

```
34     # membuat titik koordinat
35     plot(x + x_position, y + y_position)
36
37     while y > x:
38
39         if d < 0:
40             x += 1
41             d += 4 * x + 6
42         else:
43             x += 1
44             y -= 1
45             d += (4 * (x - y)) + 6
46
47         # mencari nilai (x, y)
48         # membalikkan nilai menjadi (y, x)
49
50         #nilai (x, y)
51
52         # kuadran 1
53         plot(x + x_position, y + y_position)
54
55         # kuadran 2
56         plot(x + x_position, -y + y_position)
57
58         # kuadran 3
59         plot(-x + x_position, -y + y_position)
60
61         # kuadran 4
62         plot(-x + x_position, y + y_position)
63
64         #nilai (y, x)
65
```

C: > Users > user > Downloads > bresenham dewi.py > bresenham_circle_drawing

```
66     # kuadran 1
67     plot(y + x_position, x + y_position)
68
69     # kuadran 2
70     plot(-y + x_position, x + y_position)
71
72     # kuadran 3
73     plot(-y + x_position, -x + y_position)
74
75     # kuadran 4
76     plot(y + x_position, -x + y_position)
77
78
79 def plotpoints():
80     glClear(GL_COLOR_BUFFER_BIT)
81     glColor3f(1, 1.0, 1.0)
82     glBegin(GL_LINES)
83     glVertex2f(-100, 0)
84     glVertex2f(100, 0)
85     glVertex2f(0, -100)
86     glVertex2f(0, 100)
87     glEnd()
88
89     bresenham_circle_drawing(40)
90
91     glFlush()
92
93     # menampilkan hasil output
94
95
96 def main():
97     # menampilkan hasil output
```

```

98     glutInit(sys.argv)
99     # inisialisasi tipe display glut
100    glutInitDisplayMode(GLUT_SINGLE | GLUT_RGB)
101    # inisialisasi ukuran layar glut
102    glutInitWindowSize(400, 400)
103    # inisialisasi posisi layar glut
104    glutInitWindowPosition(200, 200)
105    # inisialisasi pembuatan window
106    glutCreateWindow("Bresenham Lingkaran")
107    glutDisplayFunc(plotpoints)
108
109    init()
110    glutMainLoop()
111
112
113    main()
114

```

Output:

