

IMPLEMENTASI ALGORITMA DDA PADA PEMROGRAMAN JAVA NETBEANS

Dylan Chaniago¹, Lili Nurhalim², Rakha Aji Prayoga³, Rejun Bagaskoro⁴, Ines
Heidiani Ikasari^{5*}

Ilmu Komputer, Teknik Informatika, Universitas Pamulang, Kota Tangerang, Indonesia

Email: *dosen01374@unpam.ac.id

(* : coressponding author)

Abstrak Zaman sekarang merupakan zaman globalisasi dimana segala aspek kehidupan hampir seluruhnya telah didukung oleh bantuan teknologi yang ada. Dalam berbagai macam pengaplikasian sistem telah dibuat demi memudahkan pekerjaan sehari-hari manusia itu sendiri. Salah satunya ada pada dunia pendidikan. Dimana dunia pendidikan ini membuka celah kemajuan teknologi yang semakin pesat. Namun, tidak dapat dipungkiri pula bahwa terkadang dikalangan mahasiswa masih sulit untuk memahami terlebih menyelesaikan suatu permasalahan teknologi itu sendiri. Jenis pembelajaran yang berbeda akan memberikan pemahaman tersendiri bagi mahasiswa tersebut. Salah satunya dengan mengaplikasikan pengcodengan menggunakan Java Netbeans 8.1, JDK 1.8 untuk membentuk suatu garis demi memahami setiap masalah pada Mata Kuliah Grafika Komputer.

Kata Kunci : *Algoritma DDA, Garis, Grafika Komputer, Java Programming.*

Abstract implementation of the DDA algorithm in Java NetBeans programming. The DDA algorithm is used to draw lines in computer graphics. This research aims to help students understand the material on line formation algorithms through coding using Java NetBeans 8.1 and JDK 1.8. The research methods used are not described in the available excerpts. It is hoped that the research results can help students in their learning activities. Aulia Nurul Zahra, Mila Rosidana, et al.

Keywords: *DDA Algorithm, Lines, Computer Graphics, Java Programming.*

1. PENDAHULUAN

Seiring dengan perkembangan ilmu serta teknologi yang semakin hari semakin berkembang segala aspek kehidupan masyarakat tidak dapat terlepas dari suatu teknologi yang ada saat ini. Dari teknologi yang ada tersebutlah segala informasi dapat tersampaikan dengan baik. penyampaian informasi tersebut dapat diberikan dalam bentuk gambar, tulisan serta suara.

Pemberian informasi menggunakan gambar tentunya diperlukan suatu pembentuk dasar untuk menghasilkan gambar yang dapat dipahami sehingga menjadi suatu informasi, pembentukan dasar tersebut berasal dari garis. Namun untuk pemahaman pembentukan garis pada mata kuliah Grafika Komputer ini ada beberapa teori yang dapat dipelajari dan dipahami. Salah satu teori pembentukan garis adalah algoritma garis DDA.

Grafika komputer sendiri merupakan suatu mata kuliah wajib yang ada di Jurusan Teknik Informatika. Jurusan yang kita ketahui bahwa pemahamannya lebih dapat mudah diterima menggunakan logika. Logika yang akan digunakan pada pembentukan garis ini adalah menggunakan bahasa Java. Dengan memperhatikan beberapa hal tersebut, penulis akan mencoba membuat sebuah media pembelajaran untuk membantu mahasiswa memahami, mengetahui bagaimana suatu garis dapat dibentuk menggunakan bahasa Java dengan membuat suatu penulisan ilmiah dengan judul "Implementasi Algoritma DDA pada Pemrograman Java Netbeans" dimana tujuan dari penulisan ilmiah ini untuk menambah pengetahuan mahasiswa mengenai suatu algoritma pembentukan garis menggunakan suatu bahasa pemrograman Java.

2. METODE PENELITIAN

2.1 Algoritma Garis DDA (Digital Differential Analyzer)

Digital Differential Analyzer (DDA) adalah algoritma pembentukan garis berdasarkan perhitungan dx maupun dy . Garis dibuat dengan menentukan dua endpoint, yaitu titik awal dan titik akhir. Setiap koordinat titik yang membentuk garis diperoleh dari perhitungan, kemudian dikonversikan menjadi nilai integer. Prinsip dari Algoritma Digital Differential Analyzer (DDA) adalah mengambil nilai integer terdekat dengan jalur garis berdasarkan atas sebuah titik yang telah ditentukan sebelumnya (titik awal garis).

2.2 Titik

Grafik komputer adalah bagian dari ilmu komputer yang berkaitan dengan pembuatan dan manipulasi gambar (visual) secara digital. Untuk itu grafik sangat erat kaitannya dengan titik dan garis sebagai dasar dari manipulasi gambar.

Sebuah titik hanya dapat ditentukan oleh letaknya, tetapi tidak mempunyai ukuran (dikatakan tidak berdimensi). Sebuah titik digambarkan dengan memakai tanda noktah, kemudian dibubuhi dengan nama titik itu. Nama sebuah titik biasanya menggunakan huruf kapital seperti A, B, C, P, Q atau R. Pada gambar diperlihatkan dua titik, yaitu titik A dan titik P.

2.3 Garis

Sebuah garis (dimaksudkan adalah garis lurus) dapat diperpanjang sekehendak kita. Namun mengingat terbatasnya bidang tempat gambar, sebuah garis hanya dilukiskan sebagian saja. Bagian dari garis ini disebut wakil garis. Garis hanya mempunyai ukuran panjang, tetapi tidak mempunyai ukuran lebar. Nama dari sebuah garis dapat ditentukan dengan menyebutkan nama wakil garis itu dengan memakai huruf kecil g , h , k , atau menyebutkan nama segmen garis dari titik pangkal ke titik ujung. Pada gambar diperlihatkan dua buah garis, yaitu garis g dan segmen garis AB.

2.4 Piksel

Piksel adalah unsur gambar atau representasi sebuah titik terkecil dalam sebuah gambar grafis yang dihitung per inci.

Piksel sendiri berasal dari akronim bahasa Inggris Picture Element yang disingkat menjadi Pixel. Pada ujung tertinggi skala resolusi, mesin cetak gambar berwarna dapat menghasilkan hasil cetak yang memiliki lebih dari 2.500 titik per inci dengan pilihan 16 juta warna lebih untuk setiap inci, dalam istilah komputer berarti gambar seluas satu inci persegi yang bisa ditampilkan pada tingkat resolusi tersebut sepadan dengan 150 juta bit informasi.

Monitor yang sering kita temui terdiri dari ribuan piksel yang terbagi dalam baris-baris dan kolom-kolom. Jumlah piksel yang terdapat dalam sebuah monitor dapat kita ketahui dari resolusinya. Resolusi maksimum yang disediakan oleh monitor adalah 1024×768 , maka jumlah pixel yang ada dalam layar monitor tersebut adalah 786432 piksel. Semakin tinggi jumlah piksel yang tersedia dalam monitor, semakin tajam gambar yang mampu ditampilkan oleh monitor tersebut.

2.5 Java

Java menurut definisi dari Sun adalah nama untuk sekumpulan teknologi untuk membuat dan menjalankan perangkat lunak pada komputer standalone ataupun pada lingkungan jaringan. Java berdiri di atas sebuah mesin interpreter yang diberi nama Java Virtual Machine (JVM). Platform Java terdiri dari kumpulan library JVM, kelas-kelas loader dipaket dalam sebuah compiler,

debugger, dan perangkat lain yang dipaket dalam Java Development Kit (JDK). Untuk menjalankan file bytecode dibutuhkan JRE (Java Runtime Environment) yang memungkinkan pemakai untuk menjalankan program Java, hanya untuk menjalankan, tidak untuk membuat kode baru lagi (M. Shalahudin dan Rosa A.S, 2007).

2.6 Netbeans IDE

NetBeans adalah Integrated Development Environment (IDE) berbasis Java dari Sun Micro systems yang berjalan diatas Swing.

Swing sebuah teknologi Java untuk pengembangan aplikasi Desktop yang dapat berjalan di berbagai macam platforms seperti Windows, Linux, MacOSX and Solaris.

Netbeans merupakan software development yang Open Source, dengan kata lain software ini dibawah pengembangan bersama, bebas biaya NetBeans merupakan sebuah proyek kode terbuka yang sukses dengan pengguna yang sangat luas, komunitas yang terus tumbuh, dan memiliki hampir 100 mitra.

Suatu IDE adalah lingkup pemrograman yang diintegrasikan kedalam suatu aplikasi perangkat lunak yang menyediakan pembangun GraphicUser Interface(GUI), suatu text atau kode editor, suatu compiler atau interpreter dan suatu debugger.

3. ANALISA DAN PEMBAHASAN

Implementasi Algoritma DDA pada Pemrograman Java Netbeans adalah seperti yang dijabarkan di bawah ini, Java Netbeans yang digunakan ialah versi 8.1 dengan JDK 1.8.

Langkah pembuatan garis menggunakan Algoritma DDA yaitu dengan cara:

1. Tentukan 2 buah titik.
2. Tentukan yang menjadi titik awal (X_0, Y_0) dan titik akhir (X, Y).
3. Hitung D_x dan D_y dimana $D_x = X_1 - X_0$ dan $D_y = Y_1 - Y_0$
4. Bandingkan $Abs(D_x)$ dan $Abs(D_y)$ Jika $Abs(D_x) > Abs(D_y)$ maka $Steps = Abs(D_x)$ bila tidak $Steps = Abs(D_y)$
5. Hitung penambahan koordinat piksel, yaitu: $X_increment = D_x / steps$, dan $Y_increment = D_y / steps$.
6. Koordinat selanjutnya, yaitu $X + X_increment$ dan $Y + Y_increment$
7. Posisi piksel ditentukan dengan pembulatan nilai koordinat tersebut.
8. Ulangi langkah 6 dan 7 untuk posisi selanjutnya sampai $X = X_1, Y = Y_1$

3.1 Class garis.java

```
package DDA;
import java.awt.Graphics;
public class Garis {

    public void Garisku(Graphics g, int x0, int y0, int x1, int y1){
        int dx, dy, steps;
        int x_tambah, y_tambah, x, y;

        dx=x1-x0;
        dy=y1-y0;

        if(Math.abs(dx)> Math.abs(dy))
        {
            steps = Math.abs(dx);
        }else{
            steps = Math.abs(dy);
        }
        x_tambah=dx/steps;
        y_tambah=dy/steps;
        x=x0;
        y=y0;

        g.fillRect(x, y, 1, 1);
        for(int k=10; k<steps;k++){
            x+=x_tambah;
            y+=y_tambah;

            g.fillRect(x, y, 1, 1);
        }
    }
}
```

Gambar 1. Class Garis.java

3.2 Main Class PanggilGaris.java

```
package DDA;
import java.awt.Color;
import java.awt.Graphics;
import javax.swing.JFrame;
import javax.swing.JPanel;

public class PanggilGaris extends JPanel{
    public void paintComponent(Graphics g) {
        Garis baru = new Garis();
        g.setColor(Color.BLACK);
        baru.Garisku(g, 50, 50, 70, 100);
    }
    public static void main(String[] args){
        Garis baru=new Garis();
        JFrame frame = new JFrame("Gambar Garis");
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.add(new PanggilGaris());
        frame.setSize(400,400);
        frame.setLocationRelativeTo(null);
        frame.setVisible(true);
    }
}
```

Gambar 2. Main Class PanggilGaris.java

Pada kedua program tersebut dimasukan langkah- langkah untuk membuat garis, berikut identifikasi langkah beserta kode program yang sesuai dengan langkah-langkah tersebut :

- 1) Tentukan 2 buah titik.
Terdapat 2 buah titik yaitu x dan y
- 2) Tentukan yang menjadi titik awal (X0,Y0) dan titik akhir (X1,Y1).
- public void Garisku(Graphics g, int x0, int y0, int x1, int y1){
- baru.Garisku(g, 50, 50, 70, 100);
artinya titik awal (x0,x1) titik akhir(x1,y1) X0=50,Y0=50,X1=70,Y1=100

- 3) Hitung Dx dan Dy dimana $Dx = X1 - X0$ dan $Dy = Y1 - Y0$
 $dx = x1 - x0$; $dy = y1 - y0$; $dx = 70 - 50 = 20$; $dy = 100 - 50 = 50$;
- 4) Bandingkan $Abs(Dx)$ dan $Abs(Dy)$ Jika $Abs(Dx) > Abs(Dy)$ maka $Steps = Abs(Dx)$ bila tidak $Steps = Abs(Dy)$

```
if(Math.abs(dx) > Math.abs(dy))
{
    steps = Math.abs(dx);
} else {
    steps = Math.abs(dy);
}
```

Jika disubstitusikan :
If $(20 > 50)$ {
Steps = $Math.abs(dx)$;
}
Else { steps = $Math.abs(dy)$ }

Karena $abs(dy) > abs(dx)$ maka $steps = abs(dy) = 50$
- 5) Hitung penambahan koordinat pixel, yaitu: $X_increment = dx/steps$, dan $Y_increment = dy/steps$.
 $x_tambah = dx/steps$; $y_tambah = dy/steps$;
 $x = x0$; $y = y0$;
Jika disubstitusikan : $x_tambah = 20/50 = 0,4$; $y_tambah = 50/50 = 1$;
- 6) Koordinat selanjutnya, yaitu $X + X_increment$ dan $Y + Y_increment$
 $x += x_tambah$; $y += y_tambah$;
- 7) Posisi pixel ditentukan dengan pembulatan nilai koordinat tersebut.
- 8) Ulangi langkah 6 dan 7 untuk posisi selanjutnya sampai $X = X1$, $Y = Y1$

Pada Main Class PanggilGaris.java ditujukan untuk memanggil hasil dari perhitungan pada Class Garis.java sehingga dapat dilihat hasilnya pada Gambar 3.

3.3 Output



Gambar 3. Garis Hasil Program

Gambar diatas merupakan hasil dari perhitungan / implementasi persamaan algoritma DDA yang dijalankan menggunakan Program JavaNetbeans.

3.4 Perhitungan Secara Manual

Berdasarkan perhitungan manual seperti di bawah ini, maka akan dihasilkan Tabel 1 yaitu tabel pembentukan garis output pada Gambar 3.

Perhitungan Manual :

1. Tentukan 2 buah titik. Diketahui titik x, y
2. Tentukan yang menjadi titik awal (X_0, Y_0) dan titik akhir (X_1, Y_1)
Titik awal = A (50,50) Titik Akhir = B(70,100)
3. Hitung Dx dan Dy dimana $Dx = X_1 - X_0$ dan $Dy = Y_1 - Y_0$
 $Dx = (X_1 - X_0) = (70 - 50) = 20$ $Dy = (Y_1 - Y_0) = (100 - 50) = 50$
4. Bandingkan Abs(Dx) dan Abs(Dy) Jika Abs(Dx) > Abs(Dy) maka Steps = Abs(Dx) bila tidak Steps = Abs(Dy)
 $Abs(Dx) = Abs(20) = 20$ $Abs(Dy) = Abs(50) = 50$
 $Abs(Dy) = Abs(Dx)$
maka Steps = Abs(Dy) = 50
5. Hitung penambahan koordinat pixel, yaitu: $X_increment = dx/steps$, dan $Y_increment = dy/steps$.
 $X_increment = Dx/Steps = 20/50 = 0,4$ $Y_increment = Dy/Steps = 50/50 = 1$
6. Koordinat selanjutnya, yaitu $X + X_increment$ $Y + Y_increment$
7. Posisi pixel ditentukan dengan pembulatan nilai koordinat tersebut.
8. Ulangi langkah 6 dan 7 untuk posisi selanjutnya sampai $X = X_1, Y = Y_1$ Tabel 1. Koordinat Hasil Perhitungan

K	X	Y	Xinc	Yinc
			50	50
0	50,4	51	50	51
1	50,8	52	51	52
2	51,2	53	51	53
3	51,6	54	52	54
4	52	55	52	55
5	52,4	56	52	56
6	52,8	57	53	57
7	53,2	58	53	58
8	53,6	59	54	59
9	54	60	54	60
10	54,4	61	54	61
11	54,8	62	55	62
12	55,2	63	55	63
13	55,6	64	56	64
14	56	65	56	65

15	56,4	66	56	66
16	56,8	67	57	67
17	57,2	68	57	68
18	57,6	69	58	69
19	58	70	58	70
20	58,4	71	58	71
21	58,8	72	59	72
22	59,2	73	59	73
23	59,6	74	60	74
24	60	75	60	75
25	60,4	76	60	76
26	60,8	77	61	77
27	61,2	78	61	78
28	61,6	79	62	79
29	62	80	62	80
30	62,4	81	62	81
31	62,8	82	63	82
32	63,2	83	63	83
33	63,6	84	64	84
34	64	85	64	85
35	64,4	86	64	86
36	64,8	87	65	87
37	65,2	88	65	88
38	65,6	89	66	89
39	66	90	66	90
40	66,4	91	67	91
41	66,6	92	68	92
42	67,2	93	67	93
43	67,6	94	68	94
44	68	95	68	95
45	68,4	96	68	96
46	68,8	97	69	97

47	69,2	98	69	98
48	69,6	99	70	99
49	70	100	70	100

Keterangan

1. Karena Step = Abs(Dy)
2. Tabel :
= 50, maka dilakukan 50 langkah perhitungan dengan angka dimulai dari 0 – 49.
3. Xinc,Yinc merupakan titik piksel yang dimulai dari titik awal(50,50)
4. Koordinat selanjutnya, $X+X_increment = 50+0,4 = 50,4$ lalu $Y+Y_increment = 50+1 = 51$ begitupun seterusnya.
5. Pembulatan koordinat terjadi pada titik X.
6. Proses terus dilakukan sampai terpenuhi $X = X1, Y = Y1$, dimana $X1= 70, Y1=100$

4. KESIMPULAN

Berdasarkan penjabaran di atas dapat di Tarik kesimpulan bahwa Algoritma Digital Differential Analyzer (DDA) dapat diimplementasikan untuk pembuatan garis menggunakan Bahasa Pemrograman Java dan toolsNetBeans, kode pemrograman yang ada di dalam program tersebut sesuai dengan urutan Langkah - langkah pembuatan garis Digital Differential Analyzer(DDA).

REFERENCES

Pujiyanta, Ardi, Ir. MT., 2011, Grafika Komputer, Universitas Ahmad Dahlan, Yogyakarta.
Sutopo, Aries Hadi. Pengantar Grafika komputer. Gava Media, Yogyakarta. 2002
Agustama, M., & Handayaningsih, S. (2014). Media Pembelajaran Algoritma Garis dan Lingkaran Berbasis Multimedia. *Jurnal Sarjana Teknik Informatika*, Volume 2 Nomor 1. http://eprints.dinus.ac.id/6327/1/02_AlgoritmaGaris.pdf
Diunduh tanggal 16 Desember 2023 pukul 18.00 WIB <https://elnow.wordpress.com/2013/01/03/pengenalan-titik-dan-garis-dalam-grafik-komputer/>.
Diunduh tanggal 16 Desember 2023 pukul 19.00 WIB <https://id.wikipedia.org/wiki/Piksel>.
Diakses 16 Desember 2023 pukul 18.05 WIB <http://solidadventure07.blogspot.com/2016/02/algoritma-pembentuk-garis-pada-graphic.html?m=1>
Diakses 16 Desember 2023 pukul 22.00 WIB <http://nurmalitaca.blogspot.co.id/2017/11/hallo-perkenalkan-aku-nurmaalita-cindy.html?m=1>