



**PENERAPAN *L*-SYSTEMS DALAM MEMBANGUN
FRAKTAL *FIBONACCI WORD***

SKRIPSI

Oleh

**Reska Dian Alyagustin
NIM 091810101025**

**JURUSAN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS JEMBER**

2015



**PENERAPAN *L-SYSTEMS* DALAM MEMBANGUN
FRAKTAL *FIBONACCI WORD***

SKRIPSI

diajukan guna melengkapi tugas akhir dan memenuhi salah satu syarat
untuk penyelesaian Program Studi Matematika (S1)
dan mencapai gelar Sarjana Sains

Oleh

**Reska Dian Alyagustin
NIM 091810101025**

**JURUSAN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS JEMBER**

2015

PERSEMBAHAN

Skripsi ini saya persembahkan untuk:

1. Ibunda Komsinah dan Ayahanda Ali Wongso tercinta, yang telah membesarkan, mendidik, mendoakan, memotivasi dengan penuh kasih sayang dan perhatian yang tak pernah putus untuk putri tercintanya;
2. Nenek Aminah, Budhe Saniyah, dan Pakdhe Parno tercinta, yang telah mendoakan, memberikan semangat dan kasih sayang kepada penulis;
3. adik Muhammad Rahmatul Lil'alamin tersayang, yang telah mendoakan dan memberikan semangat dalam suka dan duka;
4. guru-guru sejak taman kanak-kanak sampai dengan perguruan tinggi;
5. Almamater Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember.

MOTTO

Hai orang-orang yang beriman, mintalah pertolongan (kepada Allah) dengan sabar dan salat; sesungguhnya Allah beserta orang-orang yang sabar.

(Terjemahan Q.S Al-Baqarah: 153) *)

Sesungguhnya dalam kesulitan ada kemudahan. Bila engkau telah selesai dari suatu pekerjaan maka kerjakan urusan yang lainnya dengan tekun.

(Terjemahan Q.S. Ash-Sarh: 6-7) **)

*) dan **) Departemen Agama Republik Indonesia. 2004. *Al-Qur'an dan Terjemahannya*. Bandung: CV Penerbit J-ART.

PERNYATAAN

Saya yang bertanda tangan dibawah ini:

nama : Reska Dian Alyagustin

NIM : 091810101025

menyatakan dengan sesungguhnya bahwa skripsi yang berjudul “Penerapan *L-Systems* dalam Membangun Fraktal *Fibonacci Word*” adalah benar-benar hasil karya sendiri, kecuali kutipan yang sudah saya sebutkan sumbernya dan belum pernah diajukan pada institusi manapun serta bukan karya jiplakan. Saya bertanggung jawab atas keabsahan dan kebenaran isinya sesuai dengan sikap ilmiah yang harus dijunjung tinggi.

Demikian pernyataan ini saya buat dengan sebenarnya, tanpa ada tekanan dan paksaan dari pihak manapun serta bersedia mendapat sanksi akademik jika ternyata dikemudian hari pernyataan ini tidak benar.

Jember, Juni 2015

Yang menyatakan,

Reska Dian Alyagustin
091810101025

SKRIPSI

**PENERAPAN *L-SYSTEMS* DALAM MEMBANGUN
FRAKTAL *FIBONACCI WORD***

Oleh

**Reska Dian Alyagustin
NIM 091810101025**

Pembimbing:

Dosen Pembimbing Utama : Kosala Dwidja Purnomo, S.Si., M.Si
Dosen Pembimbing Anggota : Kusbudiono, S.Si., M.Si

PENGESAHAN

Skripsi berjudul “Penerapan *L-Systems* dalam Membangun Fraktal *Fibonacci Word*”
telah diuji dan disahkan pada :

Hari, tanggal :

tempat : Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas
Jember.

Tim Penguji :

Ketua,

Sekretaris,

Kosala Dwidja Purnomo, S.Si., M.Si
NIP. 196908281998021001

Kusbudiono, S.Si., M.Si
NIP. 197704302005011001

Penguji I,

Penguji II,

Drs. Rusli Hidayat M.Sc
NIP. 196610121993031001

Dian Anggraeni, S.Si., M.Si
NIP. 198202162006042002

Mengesahkan
Dekan,

Prof. Drs. Kusno, DEA, Ph.D.
NIP. 196101081986021001

RINGKASAN

Penerapan *L-Systems* dalam Membangun Fraktal *Fibonacci Word*; Reska Dian Alyagustin, 091810101025; 2015; 38 halaman; Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember.

Metode *L-systems* adalah suatu metode mengganti secara bergantian bagian-bagian dari objek sederhana yang berupa segmen garis menggunakan seperangkat aturan penulisan kembali atau produksi. Objek-objek yang dibangun menggunakan *L-systems* mempunyai dua ciri utama yaitu, dapat diiterasikan sehingga membentuk beberapa generasi dan setiap bagian generasi yang terbentuk memiliki kemiripan dengan generasi sebelumnya (*self-similarity*). Kurva *Fibonacci word* merupakan salah satu objek yang memiliki kedua sifat tersebut.

Kurva *Fibonacci word* dibangun pertama kali oleh Dumaine (2009) menggunakan aturan garis ganjil-genap dari simbol barisan *Fibonacci word*. Berdasarkan bentuk kurva tersebut, Dumaine (2009) membuat suatu aksioma dan aturan produksi *L-systems* fraktal *Fibonacci word* menggunakan metode *DOL-systems*. Pada penelitian ini menggunakan aturan produksi tersebut untuk membangun fraktal *Fibonacci word* dalam dimensi dua dan mengidentifikasi perubahan fraktal *Fibonacci word* jika panjang segmen dari *string K* dan *Q* yang telah ditetapkan pada aturan produksi *L-systems* divariasikan.

Penelitian tentang penerapan *L-systems* dalam membangun fraktal *fibonacci word* ini dibagi menjadi empat tahap yaitu, penafsiran fraktal *fibonacci word* dengan metode *DOL-systems* secara matematis, penafsiran fraktal *fibonacci word* dengan metode *DOL-systems* secara grafis, pembuatan program dan analisis hasil. Aksioma dan aturan produksi *DOL-systems* pada fraktal *Fibonacci word* adalah $V = \{ L(x),$

$R(x), K(x), Q(x), K\left(\frac{x}{1+\sqrt{2}}\right), Q\left(\frac{x}{1+\sqrt{2}}\right), +, -\}$, $w = L(x)$, $p_1: L(x) \rightarrow +R(x) - L(x) K\left(\frac{x}{1+\sqrt{2}}\right) L(x) - R(x)$, $p_2: R(x) \rightarrow -L(x) + R(x) Q\left(\frac{x}{1+\sqrt{2}}\right) R(x) + L(x)$, $p_3: K(x) \rightarrow L(x)$ dan $p_4: Q(x) \rightarrow R(x)$, dengan $\theta = \pi/2$. Dari aksioma dan aturan produksi tersebut diperoleh beberapa generasi yang kemudian digunakan untuk menggambar secara grafis fraktal *fibonacci word* dengan metode *DOL-systems*. Hasil matematis dan grafis digambarkan hingga generasi ke-3 sebagai pembandingan dengan hasil visualisasi pada program. Visualisasi program fraktal *Fibonacci word* tersebut dihasilkan mulai dari generasi ke-1 sampai generasi ke-8. Berdasarkan langkah-langkah yang telah dilakukan, bentuk fraktal *Fibonacci word* yang dihasilkan dengan menerapkan metode *DOL-systems* sesuai dengan bentuk fraktal *Fibonacci word* yang dihasilkan dengan menggunakan aturan garis ganjil-genap.

Program yang telah dibuat, digunakan untuk memvariasikan nilai dari *string* K dan Q pada aturan produksi *L-systems* dengan nilai $-2l, -l, 0, \frac{l}{4}, \frac{l}{1+\sqrt{2}}, l$ dan $20l$. Untuk variasi *string* K dan $Q = -2l$, generasi ke-3 tidak memiliki pola yang sama dengan fraktal *Fibonacci word* yang belum divariasikan, namun pada generasi ke-7 secara keseluruhan memiliki pola yang sama. Pada variasi *string* K dan $Q = -l$, diperoleh bentuk baru yang polanya terlihat sama. Uniknyanya dari bentuk pola fraktal baru ini, secara matematis dapat diperoleh aturan produksi baru, sehingga memiliki dua aturan produksi. Sedangkan untuk variasi *string* K dan $Q = 0, \frac{l}{4}, \frac{l}{1+\sqrt{2}}, l$ dan $20l$ menunjukkan bahwa generasi ke-3 dan generasi ke-7 secara keseluruhan memiliki pola yang sama dengan fraktal *Fibonacci word* yang belum divariasikan.

Secara keseluruhan hasil visualisasi dari semua variasi bentuk fraktal yang diperoleh memiliki pola yang sama dengan fraktal *Fibonacci word* yang belum divariasikan. Sehingga dapat disimpulkan bahwa jika *string* L, R, K dan Q divariasikan dengan nilai yang berbeda tidak mempengaruhi pola fraktal yang dihasilkan, tetapi variasi tersebut mempengaruhi bentuk fraktal yang dihasilkan.

PRAKATA

Puji syukur penulis panjatkan kehadiran Allah SWT, yang telah melimpahkan rahmat dan hidayah-Nya sehingga penulis dapat menyelesaikan skripsi yang berjudul “Penerapan *L-Systems* dalam Membangun Fraktal *Fibonacci Word*”. Skripsi ini disusun untuk memenuhi salah satu syarat menyelesaikan pendidikan strata satu (S1) pada Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember.

Penyusunan skripsi ini tidak terlepas dari perhatian, bimbingan, motivasi dan petunjuk dari beberapa pihak, baik secara langsung maupun tidak langsung. Oleh karena itu, penulis menyampaikan terima kasih kepada:

1. Bapak Kosala Dwidja Purnomo, S.Si., M.Si, selaku Dosen Pembimbing Utama dan Bapak Kusbudiono, S.Si., M.Si, selaku Dosen Pembimbing Anggota yang dengan penuh kesabaran membimbing, mengarahkan, memberikan saran dan petunjuk dalam penyusunan skripsi ini;
2. Bapak Drs. Rusli Hidayat, M.Sc dan Ibu Dian Anggraeni, S.Si., M.Si, selaku Dosen Penguji yang telah memberikan kritik dan saran yang membangun dalam penyusunan skripsi ini;
3. seluruh staf pengajar Jurusan Matematika Fakultas MIPA Universitas Jember yang telah memberikan ilmu serta bimbingannya sehingga penulis dapat menyelesaikan skripsi ini;
4. Ibu Dian Anggraeni, S.Si., M.Si, selaku Dosen Pembimbing Akademik;
5. Ibu Komsinah, Bapak Ali Wongso, adikku Muhammad Rahmatul Lil'alamin, Nenek Aminah, Budhe Saniyah, Pakdhe Parno serta seluruh keluarga dirumah yang telah memberikan doa dan motivasi;

6. Orang-orang terbaik disekelilingku, Yunita Endah, Atik Wiwin, Achmad Fawaid, Dyah Agustinie dan Siti Izzatul, yang telah membantu, memberikan saran, memberikan motivasi, memberikan canda tawa, dan keceriaan;
7. teman-teman Matematika 2009 yang telah menemani selama menjadi mahasiswa dan memberikan semangat;
8. seluruh keluarga di kostan Kelinci 6, Tante Indri, Om Agus, Retnaningrum, Yuvi, mbak Putri, dan Jodi, yang telah membantu, memberikan semangat, memberikan canda tawa dan keceriaan;
9. serta semua pihak yang tidak dapat disebutkan satu-persatu.

Akhir kata, penulis berharap semoga skripsi ini bermanfaat dan bisa dikembangkan lagi agar lebih sempurna.

Jember, Juni 2015

Penulis

DAFTAR ISI

	Halaman
HALAMAN JUDUL	i
HALAMAN PERSEMBAHAN	ii
HALAMAN MOTTO	iii
HALAMAN PERNYATAAN	iv
HALAMAN PEMBIMBINGAN	v
HALAMAN PENGESAHAN	vi
RINGKASAN	vii
PRAKATA	ix
DAFTAR ISI	xi
DAFTAR GAMBAR	xiii
DAFTAR TABEL	xv
DAFTAR LAMPIRAN	xvi
BAB 1. PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Tujuan	3
1.4 Manfaat	3
BAB 2. TINJAUAN PUSTAKA	4
2.1 Barisan <i>Fibonacci Word</i>	4
2.2 Fraktal	5
2.2.1 Pengertian Fraktal	5
2.2.2 Fraktal <i>Fibonacci Word</i>	6
2.3 <i>Lindenmayer Systems (L-Systems)</i>	10
2.3.1 Pengertian <i>Lindenmayer Systems</i>	10

2.3.2 Jenis-jenis <i>Lindenmayer Systems</i>	12
2.3.3 Kombinasi <i>DOL-Systems</i>	14
2.3.4 Penafsiran Grafis <i>L-Sytems</i>	15
BAB 3. METODE PENELITIAN	18
3.1 Penafsiran Fraktal <i>Fibonacci Word</i> dengan Metode <i>DOL-Systems</i> secara Matematis	18
3.2 Penafsiran Fraktal <i>Fibonacci Word</i> dengan Metode <i>DOL-Systems</i> secara Grafis	19
3.3 Pembuatan Program	19
3.3 Analisis Hasil	20
BAB 4. HASIL DAN PEMBAHASAN	21
4.1 Penafsiran Fraktal <i>Fibonacci Word</i> dengan Metode <i>DOL-Systems</i> secara Matematis	21
4.2 Penafsiran Fraktal <i>Fibonacci Word</i> dengan Metode <i>DOL-Systems</i> secara Grafis	23
4.3 Pembuatan Program	25
4.3 Analisis Hasil	28
BAB 5. PENUTUP	36
5.3 Kesimpulan	36
5.3 Saran	36
DAFTAR PUSTAKA	
LAMPIRAN	

DAFTAR GAMBAR

	Halaman
Gambar 2.1 Konstruksi Fraktal <i>Fibonacci Word</i>	7
Gambar 2.2 Interpretasi <i>L-Systems</i> dalam Membangun Fraktal <i>Fibonacci word</i>	8
Gambar 2.3 $L_n = 2L_{n-3} + 2L_{n-3} \cos 90 + L_{n-6} = 2L_{n-3} + L_{n-6}$	9
Gambar 2.4 Kurva Dragon dengan $n = 10$ dan $\delta = 90^0$	14
Gambar 2.5 Pergerakan Kura-kura	16
Gambar 2.6 Penafsiran Grafis dari <i>L-Systems</i>	17
Gambar 3.1 Skema Metode Penelitian.....	18
Gambar 4.1 Generasi Pertama <i>L-Systems</i> Fraktal <i>Fibonacci Word</i>	23
Gambar 4.2 Penafsiran Fraktal <i>Fibonacci Word</i> secara Grafis Beberapa Generasi. (a) g_0 , (b) g_1 , (c) g_2 dan (d) g_3	24
Gambar 4.3 Hasil Visualisasi Fraktal <i>Fibonacci Word</i> Beberapa Generasi. (a) g_1 , (b) g_2 , (c) g_3 dan (d) g_4	26
Gambar 4.4 Hasil Visualisasi Fraktal <i>Fibonacci Word</i> Beberapa Generasi (a) g_5 , (b) g_6 , (c) g_7 dan (d) g_8	27
Gambar 4.5 Visualisasi Fraktal <i>Fibonacci Word</i> dengan Variasi Nilai <i>String</i> K dan $Q = -2l$. (a) Generasi ke-3 dan (b) Generasi ke-7.....	29
Gambar 4.6 Visualisasi Fraktal <i>Fibonacci Word</i> dengan Variasi Nilai <i>String</i> K dan $Q = -2$. (a) Generasi ke-3 dan (b) Generasi ke-7	30
Gambar 4.7 Visualisasi Fraktal <i>Fibonacci Word</i> dengan Variasi Nilai <i>String</i> K dan $Q = 0$. (a) Generasi ke-3 dan (b) Generasi ke-7	31
Gambar 4.8 Visualisasi Fraktal <i>Fibonacci Word</i> dengan Variasi Nilai <i>String</i> K dan $Q = \frac{l}{4}$. (a) Generasi ke-3 dan (b) Generasi ke-7.....	31

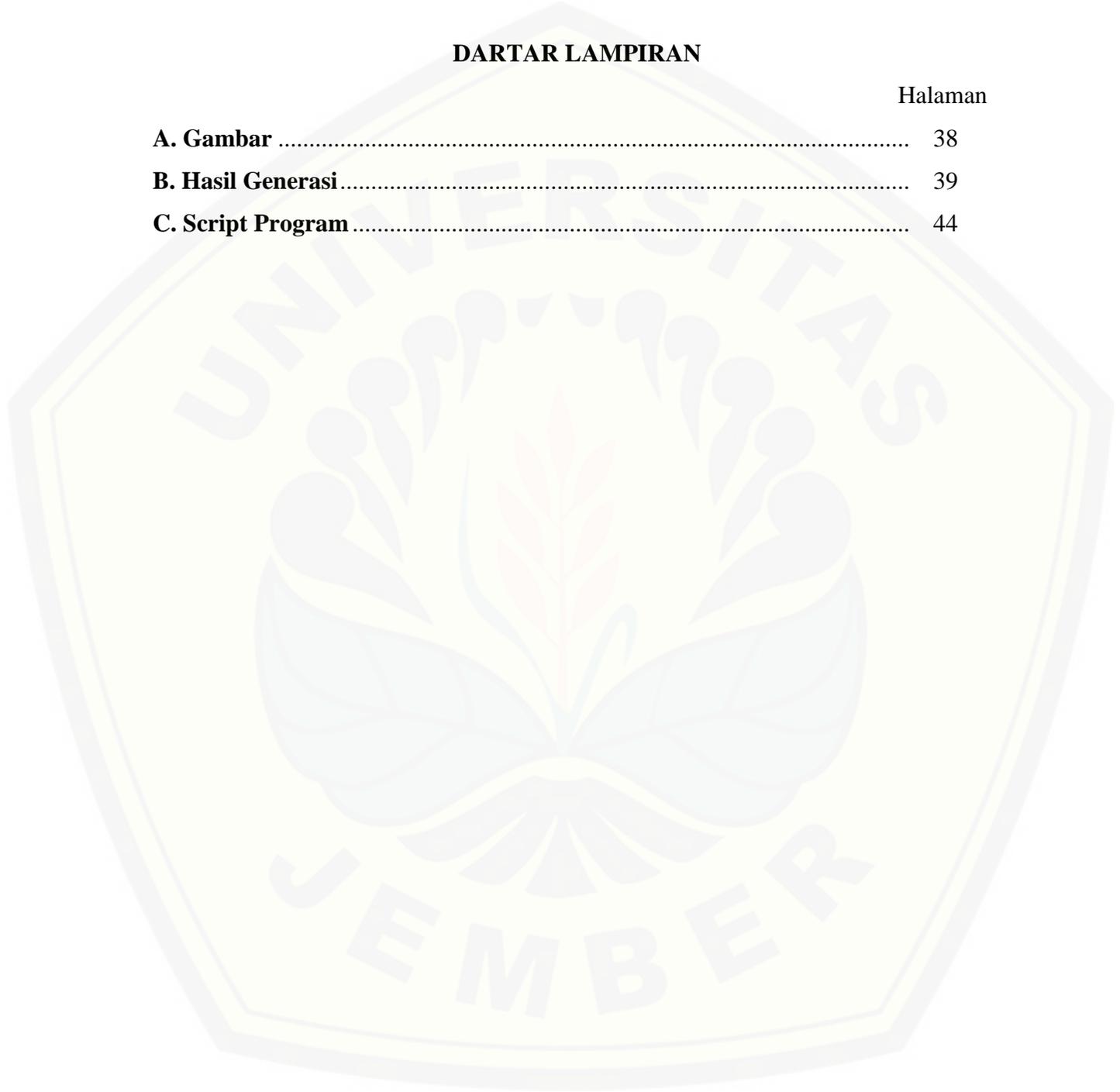
Gambar 4.9 Visualisasi Fraktal <i>Fibonacci Word</i> dengan Variasi Nilai <i>String</i> K dan $Q = l$. (a) Generasi ke-3 dan (b) Generasi ke-7	32
Gambar 4.10 Visualisasi Fraktal <i>Fibonacci Word</i> dengan Variasi Nilai <i>String</i> K dan $Q = 20l$. (a) Generasi ke-3 dan (b) Generasi ke-7	33
Gambar 4.11 Visualisasi Fraktal <i>Fibonacci Word</i> dengan Variasi Nilai <i>String</i> $L = 2l, R = l, K = Q = \frac{l}{1+\sqrt{2}}$ untuk $l = 1$. Generasi ke-3 dan (b) Generasi ke-7	34
Gambar 4.12 Visualisasi Fraktal <i>Fibonacci Word</i> dengan Variasi Nilai <i>String</i> $L = l, R = 2l, K = Q = \frac{l}{1+\sqrt{2}}$ untuk $l = 1$. Generasi ke-3 dan (b) Generasi ke-7	34

DAFTAR TABEL

	Halaman
Tabel 2.1 Generasi <i>L-Systems</i>	12
Tabel 2.2 Generasi <i>Context-Sensitive L-Systems</i>	13
Tabel 2.3 Generasi <i>Stochastic L-Systems</i>	13
Tabel 4.1 Beberapa Generasi <i>L-Systems</i> Fraktal <i>Fibonacci Word</i>	22

DARTAR LAMPIRAN

	Halaman
A. Gambar	38
B. Hasil Generasi.....	39
C. Script Program.....	44



BAB 1. PENDAHULUAN

1.1 Latar Belakang

Ilmu pengetahuan dan teknologi memberikan peranan penting dalam berbagai aspek kehidupan. Saat ini, ilmu pengetahuan dan teknologi berkembang sangat pesat terutama setelah ditemukannya komputer. Adanya komputer memudahkan manusia untuk mempelajari berbagai bidang ilmu pengetahuan. Geometri fraktal merupakan salah satu ilmu pengetahuan di bidang matematika yang berkembang cukup pesat dengan adanya komputer. Geometri fraktal ini kemudian lebih dikenal dengan istilah fraktal.

Istilah fraktal diperkenalkan oleh matematikawan kelahiran Polandia bernama Benoit B. Mandelbrot. Fraktal berasal dari bahasa Latin, yaitu kata sifat *fractus* dan kata kerja *frangere* yang berarti menguraikan menjadi kepingan-kepingan yang tak beraturan (Mandelbrot, 1977). Sedangkan geometri fraktal adalah ilmu matematika yang mendefinisikan berbagai pola tak beraturan dan terpecah-pecah serta mempelajari aspek-aspek rumit di alam sebagai suatu basis matematika (Peitgen, 1988).

Fraktal memiliki sifat *self-similarity*, yaitu setiap bagian kecil dalam sebuah fraktal dapat dipandang sebagai replikasi skala kecil dari bentuk keseluruhan. Menurut Mandelbrot (1977) fraktal dapat dibagi menjadi dua jenis, yaitu himpunan-himpunan fraktal (*fractal sets*) dan fraktal alami (*natural fractal*). Contoh fraktal jenis pertama antara lain: *Koch snowflake*, *Cantor dust*, *Sierpinski triangle*, *Julia set* dan *Mandelbrot set*. Sedangkan untuk fraktal jenis kedua, Mandelbrot mencontohkan bentuk pegunungan, awan-awan, cabang-cabang pohon, garis pantai, bongkahan salju, model fluktuasi mata uang dalam pasar finansial, dan sebagainya.

Konstruksi objek-objek fraktal pada *fractal sets* dapat dilakukan dengan beberapa cara. Salah satunya menggunakan metode *L-systems* yaitu dengan cara mengganti secara bergantian bagian-bagian dari objek sederhana yang berupa segmen garis menggunakan seperangkat aturan penulisan kembali atau produksi (Ochoa, 1998). Penelitian tentang *L-systems* telah banyak dilakukan sebelumnya yaitu oleh Nopiyanto (2006) membangun objek-objek fraktal, seperti kurva *Koch snowflake* dalam 2D menggunakan *L-systems*. Ginanjar (2010) dalam skripsinya menggunakan *L-systems* untuk membangun *Space Filling Curve* (SFC) dengan kurva Hilbert yang diperoleh dengan mempartisi interval I ke dalam 2^{2^n} subinterval yang kongruen. Mardhotillah (2012) menggunakan *L-systems* untuk membangun SFC dengan kurva Peano yang diperoleh dengan mempartisi interval I ke dalam 3^{2^n} (2D) dan 3^{3^n} (3D) subinterval yang kongruen.

Objek-objek yang dibangun menggunakan *L-systems* mempunyai dua ciri utama. Pertama yaitu objek-objek tersebut dapat diiterasikan sehingga membentuk beberapa generasi yang tujuannya untuk membangun objek-objek yang lebih kompleks dengan menggunakan sejumlah aturan produksi. Ciri yang kedua yaitu setiap bagian generasi yang terbentuk memiliki kemiripan dengan generasi sebelumnya (*self-similarity*) yang disebabkan oleh pengulangan aturan produksi yang diterapkan pada objek tersebut. Kurva *Fibonacci word* merupakan salah satu objek yang memiliki kedua sifat tersebut.

Kurva *Fibonacci word* dibangun pertama kali oleh Dumaine (2009) menggunakan aturan garis ganjil-genap dari simbol barisan *Fibonacci word*, yang kemudian diistilahkan dengan fraktal *Fibonacci word* (*Fibonacci word fractal*). Dumaine (2009), kemudian membuat suatu aksioma dan aturan produksi *L-systems* fraktal *Fibonacci word* berdasarkan bentuk kurva *Fibonacci word* dari aturan garis ganjil-genap. Berdasarkan penggunaan simbol dan aturan produksinya, aturan produksi tersebut merupakan *L-systems Context-free Deterministic (DOL-systems)*. Sehingga, penulis tertarik untuk mengkaji lebih lanjut pembangunan fraktal

Fibonacci word menggunakan metode *DOL-systems* yang telah dikemukakan oleh Dumaine (2009) dan memvisualisasikan metode tersebut menggunakan *Software Matlab*.

1.2 Rumusan Masalah

Berdasarkan uraian latar belakang tersebut, permasalahan yang akan dibahas dalam penelitian ini adalah:

- a. Bagaimana cara menerapkan metode *DOL-systems* untuk membangun fraktal *Fibonacci word* dalam dimensi dua?
- b. Bagaimana mengidentifikasi perubahan fraktal *Fibonacci word* jika panjang segmen dari *string K* dan *Q* pada aturan produksi *L-systems* divariasikan?

1.3 Tujuan

Tujuan dari penelitian ini adalah:

- a. Menerapkan metode *DOL-systems* untuk membangun fraktal *Fibonacci word* dalam dimensi dua,
- b. Membuat program untuk memvisualisasi penerapan metode *DOL-systems* dalam membangun fraktal *Fibonacci word*,
- c. Mengetahui perubahan fraktal *Fibonacci word* jika panjang segmen dari *string K* dan *Q* pada aturan produksi *L-systems* divariasikan.

1.4 Manfaat

Manfaat pembuatan program penerapan metode *DOL-systems* dalam membangun fraktal *Fibonacci word* dimensi dua adalah untuk mempermudah visualisasi fraktal *Fibonacci word*. Selain itu, program yang diperoleh dari penelitian ini dapat digunakan untuk mengidentifikasi perubahan fraktal *Fibonacci word* jika panjang segmen dari *string K* dan *Q* pada aturan produksi *L-systems* divariasikan sehingga diperoleh desain suatu pola yang lebih artistik.

BAB 2. TINJAUAN PUSTAKA

2.1 Barisan *Fibonacci Word*

Suatu barisan $(a_1, a_2, a_3, \dots, a_n, \dots)$ adalah suatu bilangan yang terurut sesuai dengan urutan bilangan asli. Barisan takhingga adalah suatu fungsi yang daerah asalnya adalah himpunan bilangan asli. Suatu barisan $(a_1, a_2, a_3, \dots, a_n, \dots)$ dapat ditulis dengan $\{a_n\}_{n=1}^{\infty}$, atau lebih singkat $\{a_n\}$ (Purcell, 1987).

Barisan *Fibonacci* diperkenalkan pada tahun 1202 oleh Leonardo Fibonacci (dikenal sebagai *Fibonacci*). Barisan ini didefinisikan secara rekursif sebagai:

$$F_n = F_{n-1} + F_{n-2} \quad (2.1)$$

dimana $F_1 = 1$ dan $F_2 = 1$ untuk $n \geq 3$. Setiap bilangan dalam barisan ini merupakan jumlah dari dua bilangan sebelumnya, sehingga untuk barisan *Fibonacci* adalah 1, 1, 2, 3, 5, 8, 13, 21, ... (Dumaine, 2009).

Barisan *Fibonacci word* adalah suatu barisan khusus dari bilangan biner (merupakan simbol dari dua huruf abjad) yang terdefinisi induktif mengikuti:

$$\begin{aligned} f_1 &= 1 \\ f_2 &= 0 \\ f_n &= f_{n-1}f_{n-2}, n \geq 3, \end{aligned} \quad (2.2)$$

dimana f_n merupakan gabungan dari dua sifat sebelumnya. Fungsi f_n dikatakan sebagai barisan berhingga dari *Fibonacci word*. Sehingga untuk

$$f = \lim_{n \rightarrow \infty} f_n = 0100101001001010010100100101 \dots \quad (2.3)$$

disebut sebagai barisan *Fibonacci word* tak hingga. Hal ini menunjukkan bahwa $|f_n| = F_n$, dimana F_n adalah bilangan barisan *Fibonacci* ke- n . Barisan *Fibonacci word* tak hingga f disebut sebagai *Sturmian word* (Dumaine, 2009). Berikut adalah Barisan *Fibonacci word* berturut-turut:

$$f_1 = 1$$

$$f_2 = 0$$

$$f_3 = 01$$

$$f_4 = 010$$

$$f_5 = 01001$$

$$f_6 = 01001010$$

$$f_7 = 0100101001001.$$

2.2 Fraktal

2.2.1 Pengertian Fraktal

Fraktal berasal dari bahasa latin, yaitu dari kata *fractus* (rusak, patah, atau tidak teratur) dan kata kerja *frangere* (memecah-mecah) yang berarti menguraikan menjadi kepingan-kepingan yang tak beraturan (Mandelbrot, 1977). Akan tetapi, dalam bentuk tak beraturan tersebut mempunyai keteraturan bentuk “kesamaan diri” (*self-similarity*) yaitu suatu bentuk yang mengulang bentuk keseluruhan dalam skala yang berbeda. Fraktal adalah suatu susunan struktur geometri yang bersifat *self-similarity* dalam berbagai skala. Terdapat tiga jenis *self-similarity* yaitu :

- a. *Exact Self-Similarity* merupakan bentuk fraktal yang tampak persis pada skala yang berbeda. Biasanya terjadi pada fraktal yang terdefinisi dengan iterasi *Iterated Function System (IFS)*, misalnya *Koch snowflake*.
- b. *Approximate Self-Similarity* merupakan bentuk fraktal yang tidak sama persis pada skala yang berbeda. Fraktal jenis ini terdiri dari bentuk yang lebih kecil dan seluruh fraktal dalam bentuk aproksimasi. *Mandelbrot set* adalah contoh fraktal yang mempunyai sifat ini.
- c. *Statistical Self-Similarity* merupakan bentuk fraktal yang tidak tervisualisasi dengan jelas tetapi memiliki ukuran statistik yang dipertahankan pada ukuran dan rasio berbeda. Contoh fraktal yang memiliki jenis ini adalah *Brownian motion* (Bourke,2002).

Karakteristik lain dari suatu fraktal adalah suatu parameter matematika yang disebut dimensi fraktal. Tidak seperti dimensi dalam geometri Euclid, dimensi fraktal pada umumnya dinyatakan dengan bilangan pecahan yang terletak di antara dua dimensi objek geometri (Peitgen, 1988). Mandelbrot (1977), Cherbit (1990) dan Falconer (1990), telah mengkaji dimensi Hausdoff pada objek fraktal. Dimensi Hausdoff dapat diaplikasikan untuk menghitung dimensi objek-objek tidak beraturan yang dimensinya tidak selalu berupa bilangan bulat, sehingga konsep dimensi Hausdoff dapat diaplikasikan untuk menghitung dimensi fraktal.

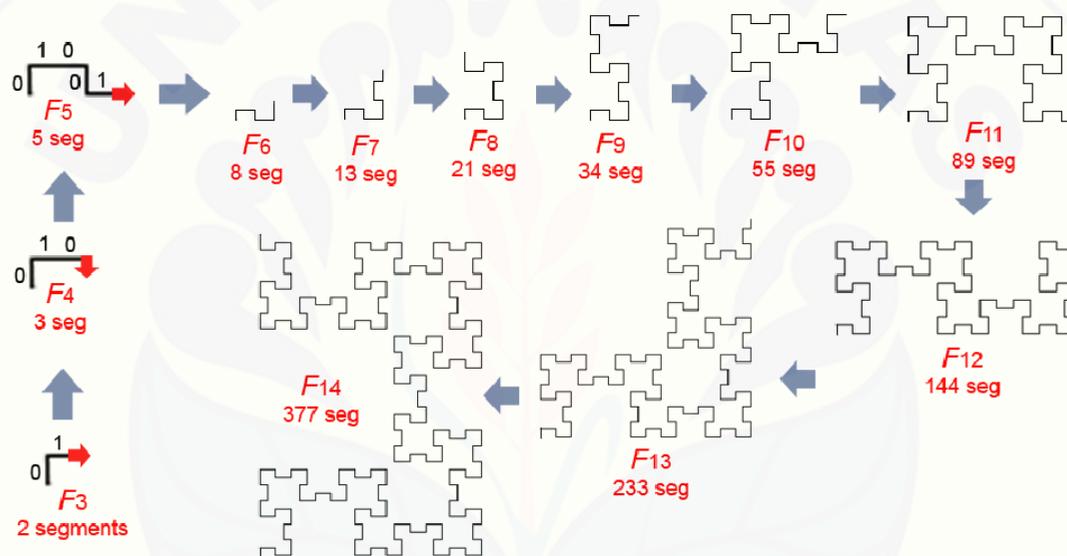
2.2.2 Fraktal *Fibonacci Word*

Fraktal *Fibonacci word* adalah suatu kurva fraktal yang memiliki sifat *self-similarity* melalui aturan gambar yang sederhana dan menarik berdasarkan pada barisan *Fibonacci word*. Barisan *Fibonacci word* dapat dihubungkan menjadi suatu kurva menggunakan gambar garis, mengikuti gerak dari simbol barisan *Fibonacci word*. Kurva ini kemudian dikenal dengan istilah fraktal *Fibonacci word*.

Aturan konstruksi fraktal *Fibonacci word* dari simbol barisan *Fibonacci word* dapat dilakukan dengan cara mengambil barisan *Fibonacci word* digit ke- n , selanjutnya gambar suatu segmen garis. Apabila digitnya "0" maka, belok kiri untuk " n " genap dan belok kanan untuk " n " ganjil, kemudian lanjutkan iterasi. Aturan konstruksi ini disebut aturan garis ganjil-genap. Garis pertama pada aturan garis ganjil-genap, dapat digambar dengan cara berikut:

- a. digit pertama adalah 0, maka gambar garis vertikal dan belok kanan,
- b. digit kedua adalah 1, maka gambar garis horizontal,
- c. digit ketiga adalah 0, maka lanjutkan menggambar garis horizontal dan belok kanan.
- d. digit keempat adalah 0, maka gambar garis vertikal dan belok kiri. Lanjutkan secara induktif (Dumaine, 2009).

Konstruksi fraktal *Fibonacci word* menggunakan aturan garis ganjil-genap dapat dilihat pada Gambar 2.1. Pada Gambar 2.1 menunjukkan pembangunan kurva fraktal, ketika aturan garis ganjil-genap diterapkan pada f_3 hingga f_{14} . Untuk selanjutnya, kurva fraktal dan *Fibonacci word* masing-masing dinamakan dengan \mathcal{F}_n dan f_n . Berdasarkan Gambar 2.1, terdapat kemiripan (*self-similarities*) antara kurva \mathcal{F}_{11} dan \mathcal{F}_8 . Hal ini menunjukkan sifat dari fraktal *Fibonacci word* dimana kurva \mathcal{F}_n memiliki kemiripan dengan \mathcal{F}_{n-3} . Dan jumlah segmen garis pada \mathcal{F}_n sama dengan jumlah bilangan barisan *Fibonacci* ke- n (F_n).



Gambar 2.1 Konstruksi Fraktal *Fibonacci Word* (Dumaine, 2009).

Selain menggunakan aturan garis ganjil-genap, konstruksi fraktal *Fibonacci word* dapat dilakukan dengan beberapa cara yaitu, aturan konstruksi dengan melibatkan *golden rasio*, *Lindenmayer Systems (L-systems)*, dengan metode menghapus menggunakan persegi putih pada persegi panjang hitam, dengan mengubah arah putaran, konstruksi menggunakan pemetaan yang berbeda, dan dengan *Rauzy ruler* (Dumaine, 2009).

Model konstruksi fraktal *Fibonacci word* dengan *L-systems* dibangun menggunakan aturan penulisan kembali garis dari metode *DOL-systems*. Dumaine

(2009) menggunakan huruf L , R , K , dan Q dalam memodelkan konstruksi fraktal *Fibonacci word*. Aturan produksi yang digunakan adalah

$$\begin{aligned}
 L(x) &\rightarrow +R(x) - L(x) K\left(\frac{x}{1+\sqrt{2}}\right) L(x) - R(x) \\
 R(x) &\rightarrow -L(x) + R(x) Q\left(\frac{x}{1+\sqrt{2}}\right) R(x) + L(x) \\
 K(x) &\rightarrow L(x) \\
 Q(x) &\rightarrow R(x)
 \end{aligned}
 \tag{2.4}$$

Simbol-simbol aturan produksi tersebut mempunyai perintah-perintah sebagai berikut:

$+$: belok kiri 90° ,

$-$: belok kanan 90° ,

$L(x)$: gambar segmen dengan panjang x , L memungkinkan pada tahap iterasi selanjutnya pembangunan kurva ke kiri,

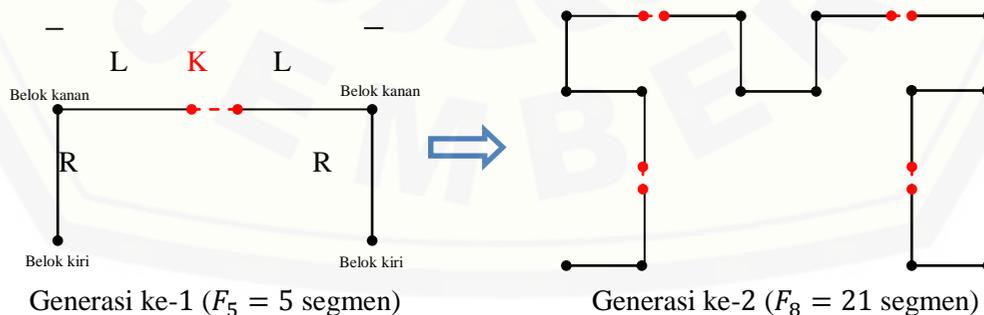
$R(x)$: gambar segmen dengan panjang x , R memungkinkan pada tahap iterasi selanjutnya pembangunan kurva ke kanan,

$K(x)$: gambar segmen dengan panjang x , menunda pembangunan L pada satu tahap iterasi,

$Q(x)$: gambar segmen dengan panjang x , menunda pembangunan R pada satu tahap iterasi,

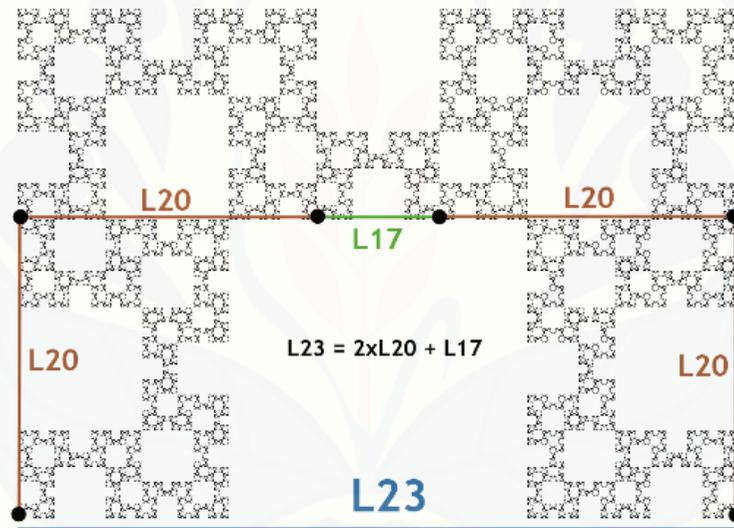
dimana panjang K dan Q adalah panjang segmen x dibagi dengan $1 + \sqrt{2}$.

Interpretasi grafik aturan produksi di atas dapat dilihat pada Gambar 2.2.



Gambar 2.2 Interpretasi L -Systems dalam Membangun Fraktal *Fibonacci Word*

Berdasarkan aturan produksi *L-systems* dan interpretasi grafik pada Gambar 2.2 dapat disimpulkan bahwa generasi pertama dari *L-systems* fraktal *Fibonacci word* adalah \mathcal{F}_5 . Diketahui jika $F_n = F_{n-1} + F_{n-2}$ maka $F_n = 4F_{n-3} + F_{n-6}$. Sehingga, kurva fraktal \mathcal{F}_n dapat dibangun dengan urutan dua kurva \mathcal{F}_{n-3} (yang dipisahkan dengan sudut persegi), kurva \mathcal{F}_{n-6} , dan dua kurva \mathcal{F}_{n-3} (yang dipisahkan dengan sudut persegi). Sehingga, untuk $\mathcal{F}_5 = 2\mathcal{F}_2 + (1 + \sqrt{2}) + 2\mathcal{F}_2$ dimana $1 + \sqrt{2}$ adalah faktor skala antara \mathcal{F}_n dan \mathcal{F}_{n-3} yang diperoleh dari perbandingan \mathcal{F}_n dan \mathcal{F}_{n-3} . Ditunjukkan bahwa \mathcal{F}_n adalah gabungan dari 4 salinan \mathcal{F}_{n-3} dan satu \mathcal{F}_{n-6} (Dumaine, 2009). Misalkan didefinisikan L_n adalah panjang fraktal f_n (lihat Gambar 2.3).



Gambar 2.3 $L_n = 2L_{n-3} + 2L_{n-3} \cos 90 + L_{n-6} = 2L_{n-3} + L_{n-6}$ (Dumaine, 2009)

Berdasarkan Gambar 2.3, disimpulkan bahwa dua salinan pertama L_{n-3} selalu orthogonal, sehingga memenuhi

$$L_n = 2L_{n-3} + 2L_{n-3} \cos 90 + L_{n-6} = 2L_{n-3} + L_{n-6} \tag{2.5}$$

Dengan definisi, faktor skala B adalah:

$$B = \frac{L_n}{L_{n-3}} = \frac{L_{n-3}}{L_{n-6}} \tag{2.6}$$

menggunakan dua identitas sebelumnya, B memenuhi:

$$\begin{aligned}
B &= \frac{L_n}{L_{n-3}} = \frac{2L_{n-3} + L_{n-6}}{L_{n-3}} \\
BL_{n-3} &= 2L_{n-3} + L_{n-6} \\
BL_{n-3} &= 2L_{n-3} + \frac{L_{n-3}}{B} \\
BL_{n-3} &= L_{n-3} \left(2 + \frac{1}{B} \right) \tag{2.7}
\end{aligned}$$

maka faktor skala B menjadi solusi persamaan:

$$\begin{aligned}
B &= \left(2 + \frac{1}{B} \right) \\
B^2 &= 2B + 1 \\
B &= 1 + \sqrt{2} \tag{2.8}
\end{aligned}$$

2.3 Lindenmayer Systems (*L-Systems*)

2.3.1 Pengertian *Lindenmayer Systems*

Lindenmayer Systems atau yang biasa disebut dengan *L-systems* diperkenalkan oleh seorang ahli biologi yang bernama Aristid Lindenmayer pada tahun 1968 sebagai dasar teori aksiomatik. *L-systems* merupakan suatu sistem dinamik simbolik dengan sifat tambahan berupa penafsiran grafis perubahan sistem, misalnya pertumbuhan dan perkembangan tanaman, pembelahan sel dan pembentukan fraktal (Wright, 1996). Prinsip dasar *L-systems* adalah suatu sistem penulisan kembali, yaitu teknik membangun objek yang kompleks dari pengulangan bagian objek yang sederhana menggunakan suatu aturan penulisan kembali yang dilakukan secara rekursif. Objek kompleks ini dibangun dengan cara mengganti bagian-bagian dari objek sederhana secara bergantian menggunakan seperangkat aturan penulisan kembali atau produksi (Ochoa, 1998).

Menurut Wright (1996), *L-systems* memiliki beberapa komponen utama yaitu huruf, aksioma dan aturan produksi. Berikut penjelasan mengenai komponen-komponen tersebut.

a. Huruf

Huruf adalah himpunan berhingga V dari simbol-simbol formal yang dapat diganti, misalnya dalam bentuk a, b, c dan seterusnya.

b. Aksioma

Aksioma (*inisiator*) adalah suatu *string* w dari simbol-simbol pada V . Himpunan *string* dari V dinotasikan V^* . Jika diberikan $V = \{a, b, c\}$, maka beberapa contoh himpunan *string* yang dapat dibentuk yaitu $V^* = \{a, b, cb, aabca, caabbbc\}$. Panjang $|w|$ dari suatu *string* w adalah jumlah simbol dalam *string*.

c. Produksi

Produksi adalah aturan penulisan kembali suatu pemetaan simbol $a \in V$ ke *string* $w \in V^*$. Pemetaan tersebut dapat dinotasikan sebagai berikut:

$$p: V \rightarrow V^*$$

$$p: a \rightarrow w$$

Jika suatu simbol $a \in V$ tidak memiliki aturan produksi, maka dapat diasumsikan bahwa simbol tersebut dipetakan pada dirinya sendiri sehingga a konstanta *L-systems* (Wright, 1996).

Misalnya diberikan komponen *L-systems* dengan $V = \{a, b\}$, $w = a$, $p_1: a \rightarrow b$ dan $p_2: b \rightarrow ba$. Hasil produksi dari *L-systems* didefinisikan sebagai barisan g_n dengan $n = 0, 1, 2, 3, \dots, k$. Untuk mendapatkan hasil produksi dari setiap generasi (g_n) yaitu dengan mengganti setiap simbol dari generasi sebelumnya g_{n-1} sesuai aturan produksinya. Aturan produksi p_1 memiliki arti a diproduksi menjadi b . Aturan produksi p_2 memiliki arti b diproduksi menjadi ba . Berikut hasil produksi generasi tersebut dapat dilihat pada Tabel 2.1.

Tabel 2.1 Generasi *L-Systems*

Generasi	Hasil Produksi
g_0	a
g_1	b
g_2	ba
g_3	bab
g_4	$babba$
g_5	$babbabab$
g_6	$babbababbabba$
g_7	$babbababbabbababbabab$
g_8	$babbababbabbababbababbabbababbabba$

2.3.2 Jenis-jenis *Lindenmayer Systems*

Jenis-jenis *L-systems* dapat dibagi berdasarkan penggunaan simbol dan aturan produksinya (Wright, 1996). Berikut penjelasan mengenai hal tersebut.

Jenis-jenis *L-systems* yang pertama adalah berdasarkan penggunaan simbol. Jenis-jenis *L-systems* ini dibagi menjadi dua yaitu *Context-free L-systems* dan *Context-sensitive L-Systems*. Keduanya diuraikan sebagai berikut.

a. *Context-free L-systems*

Context-free L-systems adalah *L-systems* yang aturan produksinya hanya memperlihatkan pada satu individu saja, bukan pada tetangga-tetangganya.

Contoh : komponen *L-systems* dengan $V = \{a, b\}$, $w = a$, $p_1: a \rightarrow b$ dan $p_2: b \rightarrow ba$.

Hasilnya ditunjukkan pada Tabel 2.1.

b. *Context-sensitive L-systems*

Context-sensitive L-systems adalah *L-systems* yang aturan produksinya untuk suatu simbol berlaku jika dan hanya jika simbol tersebut memiliki tetangga tertentu.

Contoh : komponen *L-systems* dengan $V = \{a, b\}$, $w = aaa$, $p_1: a(> a) \rightarrow ab$, $p_2: a(> b) \rightarrow c$, $p_3: c \rightarrow \emptyset$. Arti $a(> a)$ adalah jika a memiliki tetangga a disisi kanannya, maka a diproduksi menjadi ab . Hal yang sama dapat diartikan pada $a(> b)$, yaitu jika a memiliki tetangga b disisi kanannya, maka a diproduksi

menjadi c . Sehingga hasil produksi beberapa generasi selanjutnya dari sistem disajikan dalam Tabel 2.2 berikut.

Tabel 2.2 Generasi *Context-Sensitive L-Systems*

Generasi	Hasil Produksi
g_0	aaa
g_1	$ababa$
g_2	$cbcba$
g_3	bba

Kedua yaitu jenis-jenis *L-systems* berdasarkan jumlah aturan produksi. *L-systems* dibagi menjadi dua yaitu *Deterministic L-systems* dan *Stochastic L-systems* bila dilihat dari aturan produksi untuk satu simbol. Berikut penjelasan mengenai keduanya.

1) *Deterministic L-systems*

Deterministic L-systems adalah *L-systems* yang memiliki tepat satu produksi untuk setiap simbol. Suatu *L-systems Context-free Deterministic* pada umumnya disebut dengan (*DOL-systems*).

Contoh : komponen *L-systems* dengan $V = \{a, b\}$, $w = a$, $p_1: a \rightarrow b$ dan $p_2: b \rightarrow ba$.

2) *Stochastic L-systems*

Stochastic L-systems adalah *L-systems* yang memiliki lebih dari satu aturan produksi untuk satu simbol tertentu. *Stochastic L-systems* memerlukan kriteria tertentu untuk menentukan kapan aturan suatu produksi diterapkan.

Contoh : komponen *L-systems* dengan $V = \{a, b\}$, $w = abba$, $p_1: a \rightarrow b$ (untuk setiap a berada diawal suatu hasil produksi dari setiap generasi), $p_2: a \rightarrow \emptyset$ (untuk setiap a yang berada diakhir suatu hasil produksi dari setiap hasil generasi), $p_3: b \rightarrow a$. Sehingga hasil produksi beberapa generasi selanjutnya dari sistem dapat dilihat pada Tabel 2.3.

Tabel 2.3 Generasi *Stochastic L-Systems*

Generasi	Hasil Produksi
g_0	$abba$
g_1	baa
g_2	aa

2.3.3 Kombinasi *DOL-systems*

Kombinasi *DOL-systems* merupakan aturan penulisan kembali yang menggabungkan atau mengkombinasikan antara aturan penulisan kembali titik dan garis. Prusinkiewicz dan Lindenmayer (2004) menjelaskan pengertian penulisan kembali sebuah titik dan kombinasi *DOL-systems* adalah sebagai berikut.

a. Penulisan Kembali Garis

Penulisan kembali sebuah garis dapat dilihat seperti inti dari pembangunan Kurva Dragon. Misalkan pada Gambar 2.4 memperlihatkan Kurva Dragon yang dibangun menggunakan *L-systems* dengan aturan produksi:

$$\begin{aligned} F_l \\ F_l \rightarrow F_l + F_r + \\ F_r \rightarrow F_l - F_r + \end{aligned}$$

Perintah F_l dan F_r memberikan perintah menggambar sebuah garis ke depan. Aturan produksinya mengganti F_l atau F_r dan menggabungkannya dengan perintah belok kiri dengan simbol $+$ dan belok kanan dengan simbol $-$.



Gambar 2.4 Kurva Dragon dengan $n = 10$ dan $\delta = 90^\circ$

b. Penulisan Kembali Titik

Ide penulisan kembali sebuah titik adalah mengganti suatu titik dengan objek sederhana sehingga dapat membentuk objek yang lebih kompleks. Untuk membangunnya, interpretasi gerak kura-kura dikembangkan menggunakan simbol-simbol yang menggambarkan perubahan bagian-bagian yang lebih kecil.

c. Kombinasi Penulisan Garis dan Titik

Beberapa kurva dapat dibangun menggunakan penulisan kembali titik dan garis secara bersamaan. Sebagai contoh, *L-systems* yang membangun Kurva Dragon yang menggunakan penulisan kembali garis:

$$\begin{aligned} w: F_l \\ p_1: F_l \rightarrow F_l + F_r + \\ p_2: F_r \rightarrow -F_l - F_r \end{aligned}$$

Diasumsikan bahwa aturan produksi sebelumnya dapat ditulis lebih dari satu huruf, kemudian huruf subscriptnya dapat dijadikan aturan produksi sendiri. Sehingga aturannya berubah menjadi:

$$\begin{aligned} w: Fl \\ p_1: Fl \rightarrow Fl + Fr + \\ p_2: rF \rightarrow -Fl - rF \end{aligned}$$

dimana huruf l dan r diinterpretasikan oleh gerak kura-kura sebagai sebuah titik tertentu pada objek. Produksi p_1 mengganti huruf l dengan $l + Fr +$ dan dengan cara yang sama, produksi p_2 mengganti huruf r dengan $-l - rF$, sehingga aturannya menjadi:

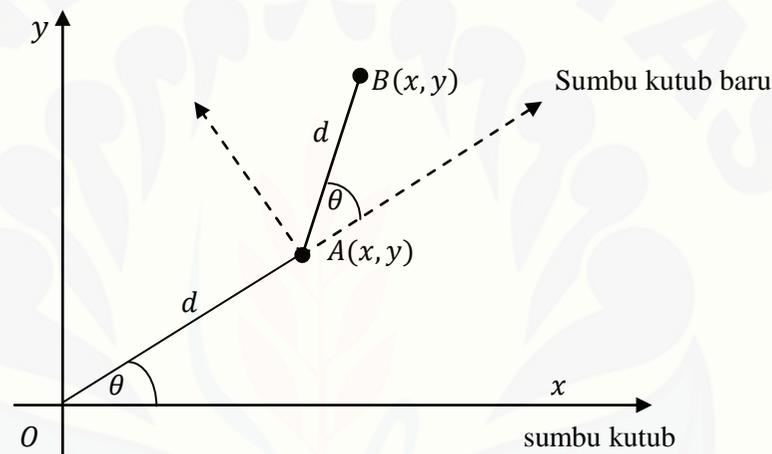
$$\begin{aligned} w: Fl \\ p_1: l \rightarrow l + Fr + \\ p_2: r \rightarrow -Fl - r \end{aligned}$$

2.3.4 Penafsiran Grafis *L-systems*

Objek-objek fraktal dibangun dan diproduksi dari suatu segmen garis berdasarkan untai (*string*) yang dihasilkan oleh *L-systems*. Untaian dari *L-systems* mengandung informasi mengenai geometri kura-kura untuk menghasilkan objek fraktal (Prunsinkiewicz, 2004).

Ide dasar dari interpretasi gerak kura-kura tersebut didefinisikan dengan posisi (x, y, θ) dimana koordinat kartesius (x, y) menggambarkan posisi awal kura-

kura dan sudut θ menunjukkan sudut awal dari arah pergerakan kura-kura. Misalkan suatu pola pergerakan kura-kura digambarkan pada suatu koordinat kartesius di R^2 dengan sumbu X positif dipandang pula sebagai sumbu kutub dari titik asal O sebagai pusat kutub. Kura-kura akan bergerak dari titik asal O sejauh d menuju ke titik $A(x, y)$ dengan arah hadap sebesar θ . Jika kura-kura bergerak kembali menuju titik $B(x, y)$ dengan arah panjang yang sama, maka titik $A(x, y)$ dianggap sebagai titik asal yang baru dan perpanjangan garis OA dianggap sebagai sumbu kutub yang baru (Gambar 2.5).



Gambar 2.5 Pergerakan Kura-kura

Pada *L-systems* terdapat simbol-simbol yang dapat ditafsirkan secara grafis. Jika diasumsikan suatu satuan panjang d dan putaran sudut θ , maka perintah-perintah dari simbol-simbol pada *L-systems* adalah sebagai berikut:

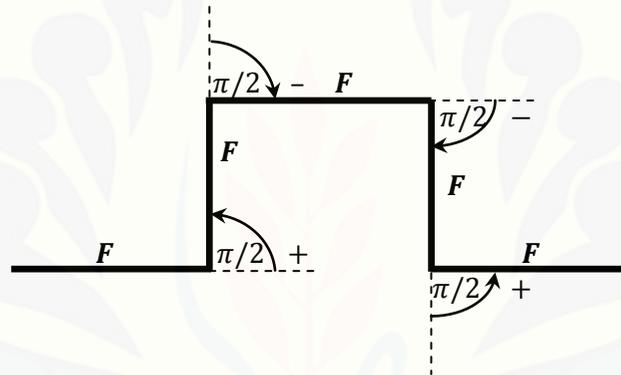
- F : menggambar ke depan satu satuan, sepanjang d . Jika diimplementasikan pada pergerakan kura-kura, dimana posisi awal kura-kura (x, y, θ) maka akan berubah menjadi (x', y', θ) dimana $x' = x + d \cos(\theta)$ dan $y' = y + d \sin(\theta)$, antara (x, y) dan (x', y') dihubungkan dengan suatu segmen garis,
- G : bergerak ke depan satu satuan, sepanjang d tanpa harus menggambar,
- $+$: berputar berlawanan arah jarum jam dengan sudut θ ,
- $-$: berputar searah jarum jam dengan sudut θ , dan

| : berputar 180° atau berbalik arah.

Penafsiran *L-systems* secara grafis dapat diartikan menggambar secara grafis barisan generasi yang dihasilkan dari aksioma dan aturan produksi yang diberikan. Contohnya, jika diberikan aksioma dan aturan produksi dengan $V = \{F, +, -\}$, $w = F$ dan $p: F \rightarrow F + F - F - F + F$, maka dimulai dengan aksioma F akan diperoleh produksi generasi pertama g_1 dengan *string*:

$$F + F - F - F + F$$

Jika diasumsikan bahwa satu satuan sudut θ adalah $\pi/2$ radian, maka penafsiran grafis dari generasi pertama dapat dilihat pada gambar berikut ini:

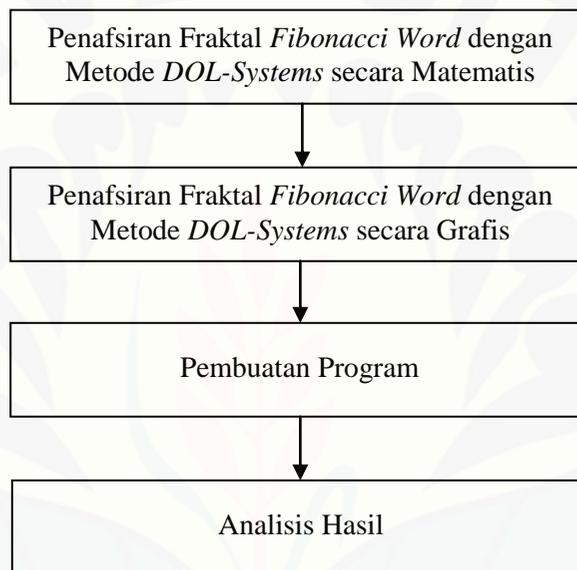


Gambar 2.6 Penafsiran Grafis dari *L-Systems*

Penafsiran grafis berdasarkan pada Gambar 2.6 tersebut mula-mula mengerjakan perintah F yaitu menggambar garis ke depan sepanjang d satuan. Perintah simbol $+$ untuk memutar arah garis berlawanan arah jarum jam sebesar $\theta = \pi/2$. Perintah berikutnya menggambar F kembali sesuai arah yang telah ditentukan pada perintah sebelumnya. Perintah simbol $-$ untuk memutar arahnya searah jarum jam sebesar $\theta = \pi/2$ dan seterusnya. Hal yang sama dapat dilakukan untuk membangun penafsiran grafis dari generasi selanjutnya.

BAB 3. METODE PENELITIAN

Langkah-langkah yang akan dilakukan untuk membangun fraktal *Fibonacci word* menggunakan metode *DOL-systems* secara skematik dapat dilihat pada Gambar 3.1.



Gambar 3.1 Skema Metode Penelitian

Dari skema pada Gambar 3.1, langkah-langkah penelitian dapat diuraikan sebagai berikut:

3.1 Penafsiran Fraktal *Fibonacci Word* dengan Metode *DOL-Systems* secara Matematis

Langkah awal dalam membangun fraktal *Fibonacci word* adalah penafsiran fraktal *Fibonacci word* dengan metode *L-systems* secara matematis. Pada langkah ini, tahap pertama yaitu menentukan komponen-komponen *L-systems* dari aturan produksi yang telah dikemukakan oleh Dumaine (2009) pada persamaan (2.4).

Aturan produksi pada persamaan (2.4) diperoleh berdasarkan bentuk fraktal *Fibonacci word* dari aturan garis ganjil genap. Aturan produksi tersebut memiliki tepat satu aturan produksi untuk setiap simbol dan aturan produksinya hanya memperlihatkan satu individu saja (tidak memiliki tetangga). Sehingga, *L-systems* fraktal *Fibonacci word* merupakan jenis *L-systems Context-free Deterministic* yang pada umumnya disebut dengan *DOL-systems*.

Kemudian berdasarkan komponen-komponen *L-systems* tersebut akan diperoleh beberapa generasi *L-systems* dari fraktal *Fibonacci word* yang akan disebutkan hingga generasi ke-3.

3.2 Penafsiran Fraktal *Fibonacci Word* dengan Metode *DOL-Systems* secara Grafis

Hasil Generasi *L-systems* fraktal *Fibonacci word* pada langkah sebelumnya, kemudian digambar secara grafis berdasarkan interpretasi gerak kura-kura sesuai dengan definisi simbol-simbol *L-systems* yang telah diuraikan sebelumnya. Langkah ini disebut penafsiran *L-systems* secara grafis yang dapat juga diartikan sebagai menggambar secara grafis barisan generasi yang dihasilkan dari aksioma dan aturan produksi yang diberikan. Penafsiran grafis fraktal *Fibonacci word* dengan metode *DOL-systems*, dibangun dengan aturan penulisan kembali garis yang akan digambarkan hingga generasi ke-3 sebagai pembanding dengan hasil visualisasi pada program.

3.3 Pembuatan Program

Langkah selanjutnya yaitu pembuatan program visualisasi fraktal *Fibonacci word* berdasarkan hasil penafsiran fraktal *Fibonacci word* dengan metode *DOL-systems* secara matematis dan grafis. Algoritma program penerapan *L-systems* dalam membangun fraktal *Fibonacci word* diuraikan sebagai berikut:

- a. Menentukan aksioma dan aturan produksi;

- b. Menentukan nilai generasi fraktal *Fibonacci word* yang akan divisualisasi sebagai input. Nilai generasi tersebut dinyatakan dalam bilangan positif dan dimulai dari angka satu;
- c. Menentukan panjang segmen, sudut putar, arah dan posisi titik awal pada fraktal *Fibonacci word*, yaitu:
 - 1) panjang segmen garis pada fraktal *Fibonacci word*:
untuk $L(x)$ dan $R(x)$, $x = l$ satuan panjang
untuk $K(x)$ dan $Q(x)$, $x = \frac{l}{1+\sqrt{2}}$ satuan panjang;
 - 2) nilai satu satuan sudut θ (sudut putar) adalah $\pi/2$ radian;
 - 3) perintah belok kanan dan belok kiri dalam program menggunakan asumsi empat arah yaitu utara, selatan, timur dan barat;
 - 4) posisi titik awal adalah $(X_0, Y_0) = (0,0)$, menghadap kearah timur atau utara.
- d. Mengiterasikan nilai generasi hingga generasi ke- n berdasarkan aksioma dan aturan produksi yang telah diberikan;
- e. Menggambar fraktal *Fibonacci word* berdasarkan ketentuan pada point c dan generasi yang telah didapatkan pada langkah sebelumnya.

3.4 Analisis Hasil

Hasil yang diperoleh dari pembuatan program adalah visualisasi fraktal *Fibonacci word* dalam dimensi dua. Selanjutnya, program yang telah dibuat tersebut digunakan untuk memvariasikan panjang segmen dari *string* K dan Q pada aturan produksi *L-systems* dengan nilai $-2l$, $-l$, 0 , $\frac{l}{4}$, l dan $20l$ sehingga dapat diketahui perubahan pola fraktal *Fibonacci word*.

BAB 4. HASIL DAN PEMBAHASAN

Pada bab ini akan dibahas tentang pembangunan fraktal *Fibonacci word* menggunakan metode *DOL-systems* berdasarkan langkah-langkah yang telah diuraikan pada bab 3.

4.1 Penafsiran Fraktal *Fibonacci Word* dengan Metode *DOL-Systems* secara Matematis

Penafsiran fraktal *Fibonacci word* secara matematis dibangun dengan menentukan komponen-komponen *L-systems* dalam dimensi dua. Seperti yang telah disebutkan oleh Dumaine (2009) pada persamaan (2.4) dan Wright (1996) mengenai komponen utama *L-systems*, maka komponen-komponen *L-systems* pada fraktal *Fibonacci word* adalah

$$\begin{aligned}
 V &= \{L(x), R(x), K(x), Q(x), K\left(\frac{x}{1+\sqrt{2}}\right), Q\left(\frac{x}{1+\sqrt{2}}\right), +, -\}, \\
 w &= L(x), \\
 p_1: L(x) &\rightarrow +R(x) - L(x) K\left(\frac{x}{1+\sqrt{2}}\right) L(x) - R(x), \\
 p_2: R(x) &\rightarrow -L(x) + R(x) Q\left(\frac{x}{1+\sqrt{2}}\right) R(x) + L(x), \\
 p_3: K(x) &\rightarrow L(x), \\
 p_4: Q(x) &\rightarrow R(x)
 \end{aligned} \tag{4.1}$$

dimana simbol $+$ adalah perintah belok kiri atau memutar arah garis berlawanan arah jarum jam sebesar $\theta = \pi/2$. Sedangkan simbol $-$, adalah perintah belok kanan atau memutar arah garis searah jarum jam sebesar $\theta = \pi/2$.

Selanjutnya, persamaan (4.1) disebut dengan *L-systems* fraktal *Fibonacci word*. Berdasarkan *L-systems* fraktal *Fibonacci word* tersebut diperoleh generasi *L-systems* fraktal *Fibonacci word* yang dapat dilihat pada tabel 4.1. Generasi fraktal

Fibonacci word dibangun berdasarkan urutan pembangunan kurva *Fibonacci word* (\mathcal{F}_n) dari aturan garis ganjil genap, yaitu $\mathcal{F}_n = 2\mathcal{F}_{n-3} + \mathcal{F}_{n-6} + 2\mathcal{F}_{n-3}$ (simbol penjumlahan merupakan penulisan secara berurutan dari fraktal *Fibonacci word*). Beberapa generasi *L-systems* fraktal *Fibonacci word* pada tabel 4.1 tersebut digunakan sebagai pembandingan dari hasil generasi *L-systems* fraktal *Fibonacci word* menggunakan algoritma program.

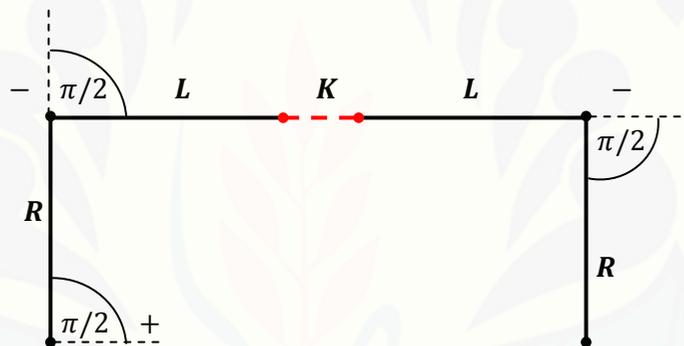
Tabel 4.1 Beberapa Generasi *L-Systems* Fraktal *Fibonacci Word*

Generasi	Hasil Produksi
g_0	L
g_1	$+R(x) - L(x) K\left(\frac{x}{1+\sqrt{2}}\right) L(x) - R(x)$
g_2	$-L(x) + R(x)Q\left(\frac{x}{1+\sqrt{2}}\right) R(x) + L(x) - R(x) - L(x) K\left(\frac{x}{1+\sqrt{2}}\right) L(x) - R(x) + L(x)$ $+R(x) - L(x) K\left(\frac{x}{1+\sqrt{2}}\right) L(x) - R(x) - L(x) + R(x)Q\left(\frac{x}{1+\sqrt{2}}\right) R(x) + L(x)$
g_3	$+R(x) - L(x) K\left(\frac{x}{1+\sqrt{2}}\right) L(x) - R(x) + L(x) + R(x)Q\left(\frac{x}{1+\sqrt{2}}\right) R(x) + L(x) - R(x)$ $-L(x) + R(x)Q\left(\frac{x}{1+\sqrt{2}}\right) R(x) + L(x) + R(x) - L(x) K\left(\frac{x}{1+\sqrt{2}}\right) L(x) - R(x)$ $-L(x) + R(x)Q\left(\frac{x}{1+\sqrt{2}}\right) R(x) + L(x) - R(x) - L(x) K\left(\frac{x}{1+\sqrt{2}}\right) L(x) - R(x) + L(x)$ $+R(x) - L(x) K\left(\frac{x}{1+\sqrt{2}}\right) L(x) - R(x) - L(x) + R(x)Q\left(\frac{x}{1+\sqrt{2}}\right) R(x) + L(x)$ $+R(x) - L(x) K\left(\frac{x}{1+\sqrt{2}}\right) L(x) - R(x)$ $+L(x) + R(x)Q\left(\frac{x}{1+\sqrt{2}}\right) R(x) + L(x) - R(x) - L(x) K\left(\frac{x}{1+\sqrt{2}}\right) L(x) - R(x) + L(x)$ $+R(x) - L(x) K\left(\frac{x}{1+\sqrt{2}}\right) L(x) - R(x) - L(x) + R(x)Q\left(\frac{x}{1+\sqrt{2}}\right) R(x) + L(x)$ $-R(x) - L(x) K\left(\frac{x}{1+\sqrt{2}}\right) L(x) - R(x) + L(x) + R(x)Q\left(\frac{x}{1+\sqrt{2}}\right) R(x) + L(x) - R(x)$ $-L(x) + R(x)Q\left(\frac{x}{1+\sqrt{2}}\right) R(x) + L(x) + R(x) - L(x) K\left(\frac{x}{1+\sqrt{2}}\right) L(x) - R(x)$

4.2 Penafsiran Fraktal *Fibonacci Word* dengan Metode *DOL-Systems* secara Grafis

Generasi *L-systems* fraktal *Fibonacci word* pada tabel 4.1, selanjutnya digambar secara grafis berdasarkan interpretasi gerak kura-kura sesuai dengan definisi simbol-simbol *L-systems*. Langkah ini disebut dengan penafsiran *L-systems* fraktal *Fibonacci word* secara grafis. Penafsiran grafis fraktal *Fibonacci word* dengan metode *DOL-systems*, dibangun dengan aturan penulisan kembali garis.

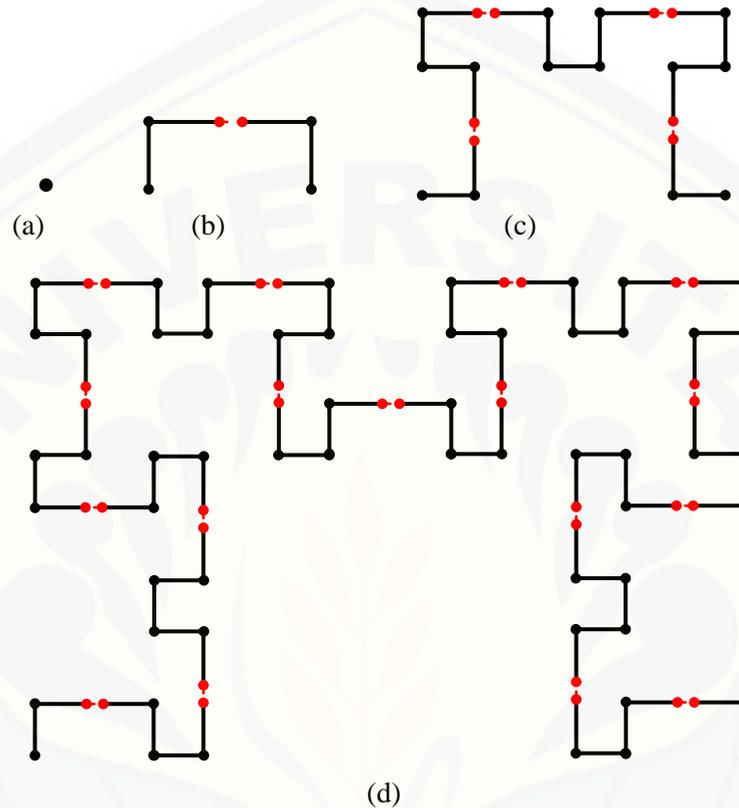
Jika diasumsikan bahwa satu satuan sudut θ adalah $\pi/2$ radian, maka penafsiran grafis generasi pertama *L-systems* fraktal *Fibonacci word* dapat dilihat pada gambar berikut ini:



Gambar 4.1 Generasi Pertama *L-Systems* Fraktal *Fibonacci Word*

Penafsiran grafis berdasarkan pada Gambar 4.1 tersebut mula-mula mengerjakan perintah simbol + untuk memutar arah garis berlawanan arah jarum jam sebesar $\theta = \pi/2$. Perintah **R** yaitu menggambar garis ke depan sepanjang l satuan sesuai arah yang telah ditentukan pada perintah sebelumnya. Perintah simbol - untuk memutar arahnya searah jarum jam sebesar $\theta = \pi/2$. Perintah **L** menggambar kembali garis ke depan sepanjang l satuan sesuai arah yang telah ditentukan pada perintah sebelumnya. Perintah berikutnya **K** menggambar kembali garis ke depan sepanjang $\frac{l}{1+\sqrt{2}}$ satuan dengan arah yang sama dengan perintah sebelumnya dan seterusnya. Hal yang sama dapat dilakukan untuk membangun penafsiran grafis dari

generasi selanjutnya. Sehingga penafsiran fraktal *Fibonacci word* secara grafis mulai dari generasi g_0 , g_1 , g_2 dan g_3 pada tabel 4.1 dapat dilihat pada Gambar 4.2.



Gambar 4.2 Penafsiran Fraktal *Fibonacci Word* secara Grafis Beberapa Generasi. (a) g_0 , (b) g_1 , (c) g_2 dan (d) g_3 .

Penafsiran grafis fraktal *Fibonacci word* dengan metode *DOL-systems* pada Gambar 4.2 digambarkan hingga generasi ke-3 sebagai pembandingan dengan hasil visualisasi pada program.

Dari penafsiran matematis dan grafis fraktal *Fibonacci word* dengan metode *DOL-systems* serta uraian pada bab 2, dapat dihitung jumlah segmen garis setiap generasi *L-systems* fraktal *Fibonacci word*. Seperti yang telah disebutkan pada bab 2, generasi pertama dari *L-systems* fraktal *Fibonacci word* adalah \mathcal{F}_5 , dimana \mathcal{F}_5 merupakan kurva fraktal dari bilangan *Fibonacci word* ke-5 (f_5). Diketahui $|f_n| = F_n$, dimana F_n adalah bilangan barisan *Fibonacci* ke- n . Berdasarkan kedua hal

tersebut dapat disimpulkan bahwa jumlah segmen garis dari kurva \mathcal{F}_n sama dengan jumlah bilangan barisan *Fibonacci* ke- n (F_n). Sehingga jika generasi ke-2 sama dengan kurva \mathcal{F}_8 , maka jumlah segmen garis generasi ke-2 *L-systems* fraktal *Fibonacci word* sama dengan jumlah bilangan barisan *Fibonacci* ke-8 yaitu 21.

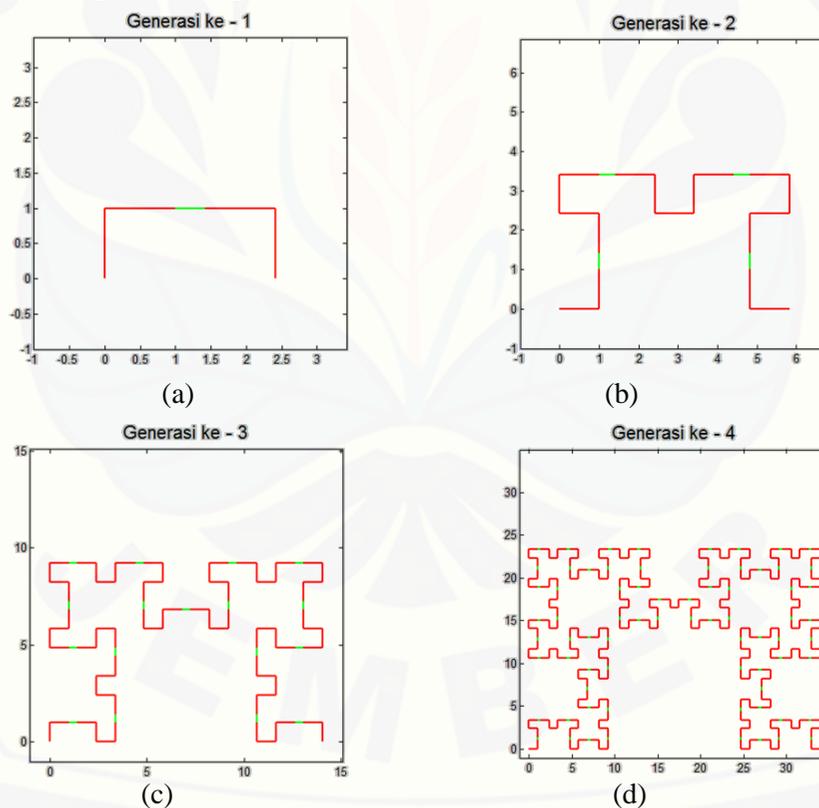
4.3 Pembuatan Program

Langkah selanjutnya yaitu pembuatan program visualisasi fraktal *Fibonacci word* dengan metode *DOL-systems* dalam dimensi dua berdasarkan hasil penafsiran fraktal *Fibonacci word* secara matematis dan grafis. Program visualisasi fraktal *Fibonacci word* dimensi dua dilakukan dengan menggunakan *Software Matlab* dan *script* program terdapat dalam Lampiran C. Algoritma program penerapan metode *DOL-systems* dalam membangun fraktal *Fibonacci word* diuraikan sebagai berikut:

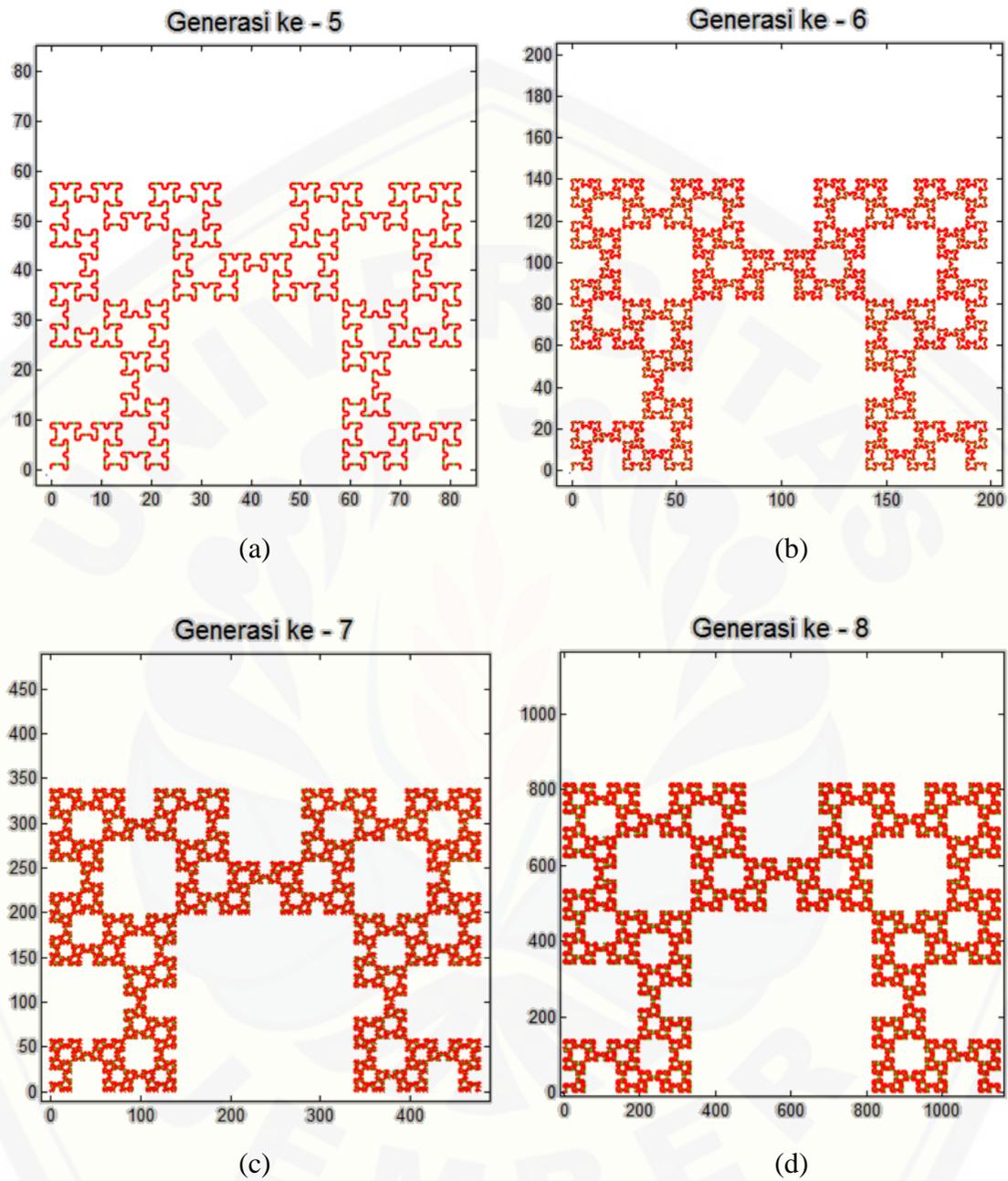
- a. Menentukan aksioma dan aturan produksi;
- b. Menentukan nilai generasi fraktal *Fibonacci word* yang akan divisualisasi sebagai input. Nilai generasi tersebut dinyatakan dalam bilangan positif dan dimulai dari angka satu;
- c. Menentukan panjang segmen, sudut putar, arah dan posisi titik awal pada fraktal *Fibonacci word*, yaitu:
 - 1) panjang segmen garis pada fraktal *Fibonacci word*:
untuk $L(x)$ dan $R(x)$, $x = l$ satuan panjang
untuk $K(x)$ dan $Q(x)$, $x = \frac{l}{1+\sqrt{2}}$ satuan panjang;
 - 2) nilai satu satuan sudut θ (sudut putar) adalah $\pi/2$ radian;
 - 3) perintah belok kanan dan belok kiri dalam program menggunakan asumsi empat arah yaitu utara, selatan, timur dan barat;
 - 4) posisi titik awal adalah $(X_0, Y_0) = (0,0)$, menghadap kearah timur atau utara.
- d. Mengiterasikan nilai generasi hingga generasi ke- n berdasarkan aksioma dan aturan produksi yang telah diberikan;

- e. Menggambar fraktal *Fibonacci word* berdasarkan ketentuan pada point c dan generasi yang telah didapatkan pada langkah sebelumnya.

Dengan menjalankan program didapatkan hasil visualisasi fraktal *Fibonacci word* dengan metode *DOL-systems* dalam beberapa generasi. Visualisasi fraktal *Fibonacci word* tersebut dihasilkan mulai dari generasi ke-1 sampai generasi ke-8, hal ini dikarenakan keterbatasan prosesor dan memori pada komputer yang digunakan. Hasil visualisasi fraktal *Fibonacci word* dimulai dari generasi ke-1 sampai generasi ke-4 dapat dilihat pada Gambar 4.3 dengan panjang $L = l$ satuan panjang, $R = l$ satuan panjang, $K = \frac{l}{1+\sqrt{2}}$ satuan panjang, dan $Q = \frac{l}{1+\sqrt{2}}$ satuan panjang, untuk $l = 1$ satuan panjang. Sedangkan untuk generasi ke-5 sampai generasi ke-8 dapat dilihat pada Gambar 4.4.



Gambar 4.3 Hasil Visualisasi Fraktal *Fibonacci Word* Beberapa Generasi. (a) g_1 , (b) g_2 , (c) g_3 dan (d) g_4 .



Gambar 4.3 Hasil Visualisasi Fraktal *Fibonacci Word* Beberapa Generasi. (a) g_5 , (b) g_6 , (c) g_7 dan (d) g_8 .

Hasil visualisasi program fraktal *Fibonacci word* dengan metode *DOL-systems* pada Gambar 4.3 dan 4.4, memiliki bentuk sama dengan hasil penafsiran fraktal *Fibonacci word* dengan metode *DOL-systems* secara grafis. Demikian juga dengan hasil generasi yang diperoleh dari program. Hasil generasi yang diperoleh dari program memiliki urutan yang sama dengan hasil penafsiran fraktal *Fibonacci word* dengan metode *DOL-systems* secara matematis. Untuk hasil generasi yang diperoleh dari program dari generasi ke-1 sampai ke-8 dapat dilihat pada lampiran B.

Berdasarkan langkah-langkah yang telah dilakukan, yaitu penafsiran fraktal *Fibonacci word* dengan metode *DOL-systems* secara matematis, grafis dan pembuatan program, maka metode *DOL-systems* dapat diterapkan dalam membangun fraktal *Fibonacci word*. Bentuk fraktal *Fibonacci word* yang dihasilkan dengan menerapkan metode *DOL-systems* sesuai dengan bentuk fraktal *Fibonacci word* yang dihasilkan dengan menggunakan aturan garis ganjil-genap.

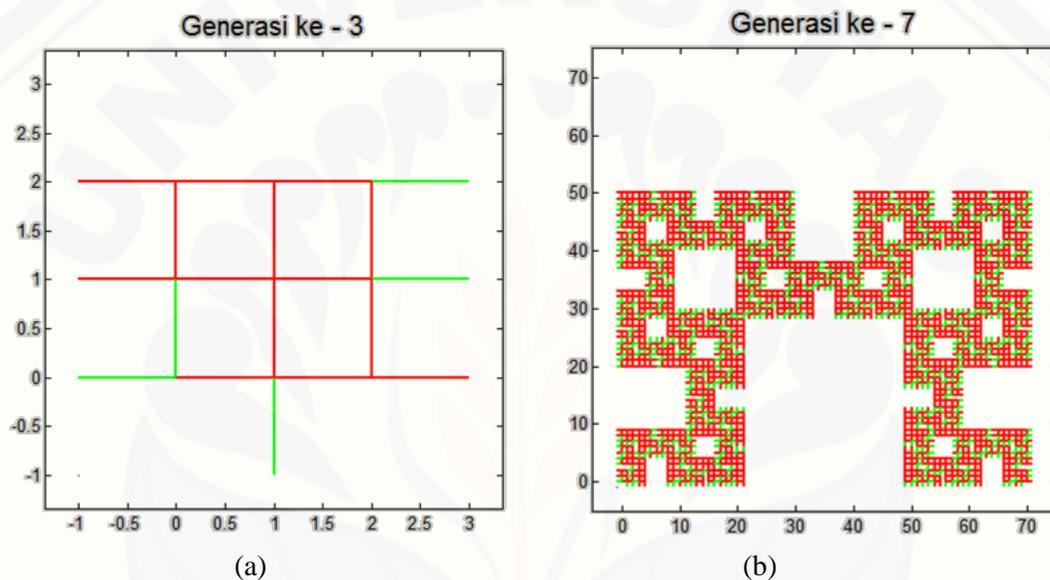
4.4 Analisis Hasil

Pada subbab 4.3 telah dibuat program visualisasi fraktal *Fibonacci word* dalam dimensi dua dengan aturan produksi *DOL-systems* yang dimukakan oleh Dumaine (2009), yaitu nilai dari panjang *string* $L = l$ satuan panjang, $R = l$ satuan panjang, $K = \frac{l}{1+\sqrt{2}}$ satuan panjang, dan $Q = \frac{l}{1+\sqrt{2}}$ satuan panjang. Selanjutnya, program yang telah dibuat tersebut digunakan untuk memvariasikan panjang segmen dari *string* K dan Q pada aturan produksi *L-systems* dengan nilai $-2l$, $-l$, 0 , $\frac{l}{4}$, l dan $20l$ sehingga dapat diketahui perubahan pola fraktal *Fibonacci word*. Diharapkan dengan memvariasi panjang segmen dari *string* K dan Q pada aturan produksi *L-systems* dengan nilai $-2l$, $-l$, 0 , $\frac{l}{4}$, l dan $20l$, pola fraktal yang diperoleh memiliki pola sama dengan pola fraktal yang belum divariasikan yaitu panjang *string* $L = l$ satuan panjang, $R = l$ satuan panjang, $K = \frac{l}{1+\sqrt{2}}$ satuan panjang, dan $Q = \frac{l}{1+\sqrt{2}}$

satuan panjang. Berikut visualisasi hasil variasi nilai $-2l$, $-l$, 0 , $\frac{l}{4}$, l dan $20l$ dari panjang segmen *string* K dan Q .

- a. Visualisasi fraktal *Fibonacci word* dengan variasi nilai K dan $Q = -2l$

Hasil visualisasi fraktal *Fibonacci word* dengan panjang *string* $L = l$ satuan panjang, $R = l$ satuan panjang, $K = -2l$ satuan panjang, dan $Q = -2l$ satuan panjang untuk $l = 1$ dapat dilihat pada Gambar 4.5.



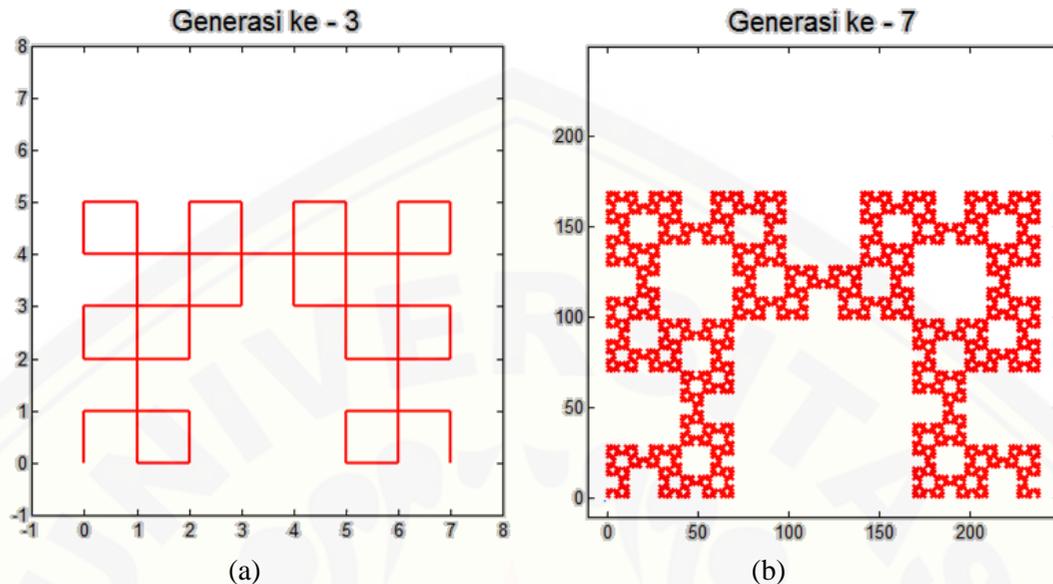
Gambar 4.5 Visualisasi Fraktal *Fibonacci Word* dengan Variasi Nilai *String* K dan $Q = -2l$.

(a) Generasi ke-3 dan (b) Generasi ke-7

Generasi ke-3 pada Gambar 4.5 tidak memiliki pola yang sama dengan fraktal *Fibonacci word* yang belum divariasikan, namun pada generasi ke-7 secara keseluruhan memiliki pola yang sama.

- b. Visualisasi fraktal *Fibonacci word* dengan variasi nilai K dan $Q = -l$

Hasil visualisasi fraktal *Fibonacci word* dengan panjang *string* $L = l$ satuan panjang, $R = l$ satuan panjang, $K = -l$ satuan panjang, dan $Q = -l$ satuan panjang untuk $l = 1$ dapat dilihat pada Gambar 4.6.



Gambar 4.6 Visualisasi Fraktal *Fibonacci Word* dengan Variasi Nilai *String K* dan $Q = -l$.

(a) Generasi ke-3 dan (b) Generasi ke-7

Generasi ke-3 pada Gambar 4.6 membentuk pola fraktal baru yang terlihat sama dengan pola fraktal *Fibonacci word* yang belum divariasikan. Sedangkan generasi ke-7 pada Gambar 4.6 secara keseluruhan memiliki pola yang sama. Secara matematis dalam algoritma program, Gambar 4.6 menggunakan aturan produksi:

$$\begin{aligned}
 L(x) &\rightarrow +R(x) - L(x) K(-x) L(x) - R(x) \\
 R(x) &\rightarrow -L(x) + R(x) Q(-x) R(x) + L(x)
 \end{aligned}
 \tag{4.1}$$

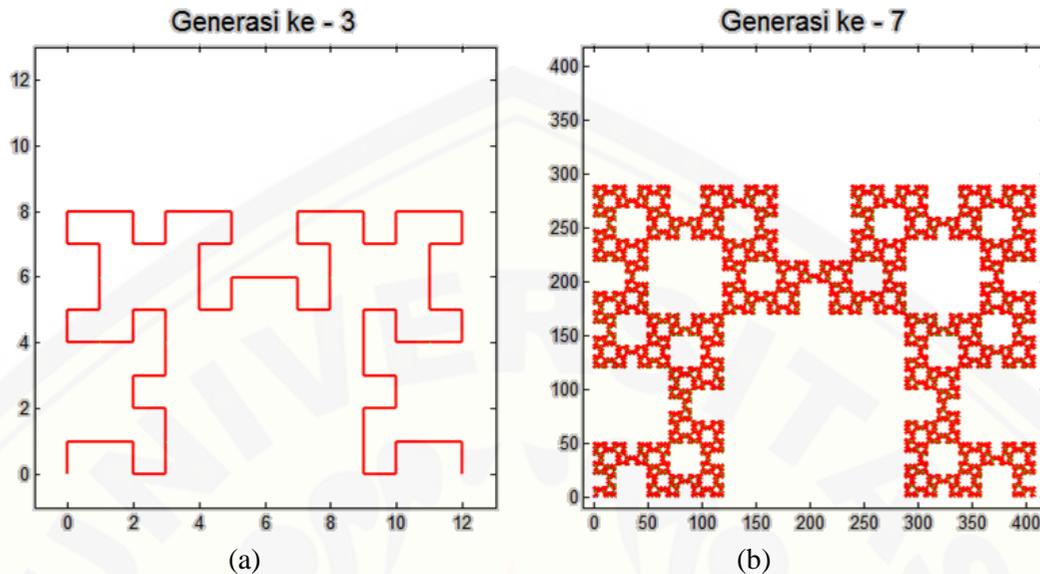
dimana $l = 1$. Akan tetapi, berdasarkan hasil visualisasi bentuk pola fraktal pada Gambar 4.6 secara matematis membentuk suatu aturan produksi baru yaitu:

$$\begin{aligned}
 L(x) &\rightarrow +R(x) - L(x) - R(x) \\
 R(x) &\rightarrow -L(x) + R(x) + L(x)
 \end{aligned}
 \tag{4.2}$$

Sehingga, dapat disimpulkan Gambar 4.6 memiliki dua aturan produksi.

c. Visualisasi fraktal *Fibonacci word* dengan variasi nilai K dan $Q = 0$

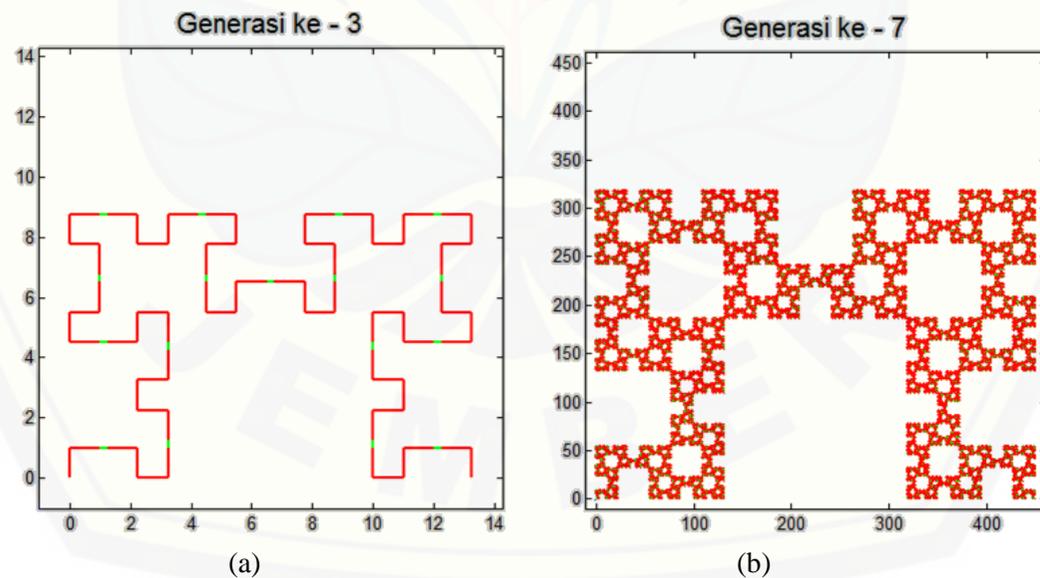
Hasil visualisasi fraktal *Fibonacci word* dengan panjang *string* $L = l$ satuan panjang, $R = l$ satuan panjang, $K = 0$ satuan panjang, dan $Q = 0$ satuan panjang untuk $l = 1$ dapat dilihat pada Gambar 4.7.



Gambar 4.7 Visualisasi Fraktal *Fibonacci Word* dengan Variasi Nilai *String K* dan $Q = 0$.
 (a) Generasi ke-3 dan (b) Generasi ke-7

Berdasarkan Gambar 4.7, generasi ke-3 dan generasi ke-7 secara keseluruhan memiliki pola yang sama dengan fraktal *Fibonacci word* yang belum divariasikan.

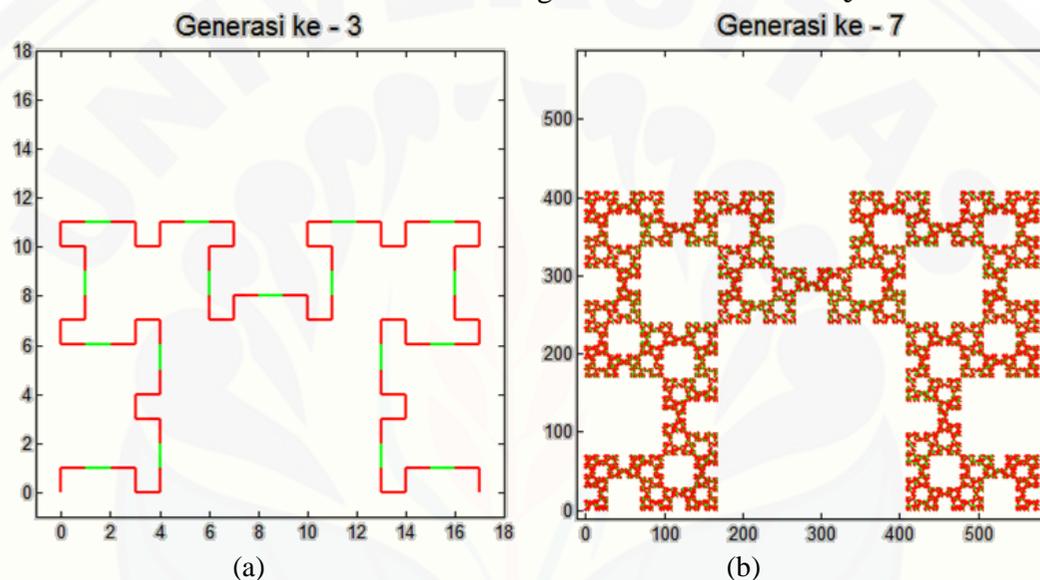
d. Visualisasi fraktal *Fibonacci word* dengan variasi nilai K dan $Q = l/4$



Gambar 4.8 Visualisasi Fraktal *Fibonacci Word* dengan Variasi Nilai *String K* dan $Q = \frac{l}{4}$.
 (a) Generasi ke-3 dan (b) Generasi ke-7

Pada Gambar 4.8 dapat dilihat hasil visualisasi fraktal *Fibonacci word* dengan panjang *string* $L = l$ satuan panjang, $R = l$ satuan panjang, $K = \frac{l}{4}$ satuan panjang, dan $Q = \frac{l}{4}$ satuan panjang untuk $l = 1$. Berdasarkan Gambar 4.8, generasi ke-3 dan generasi ke-7 secara keseluruhan memiliki pola yang sama dengan fraktal *Fibonacci word* yang belum divariasikan.

e. Visualisasi fraktal *Fibonacci word* dengan variasi nilai K dan $Q = l$

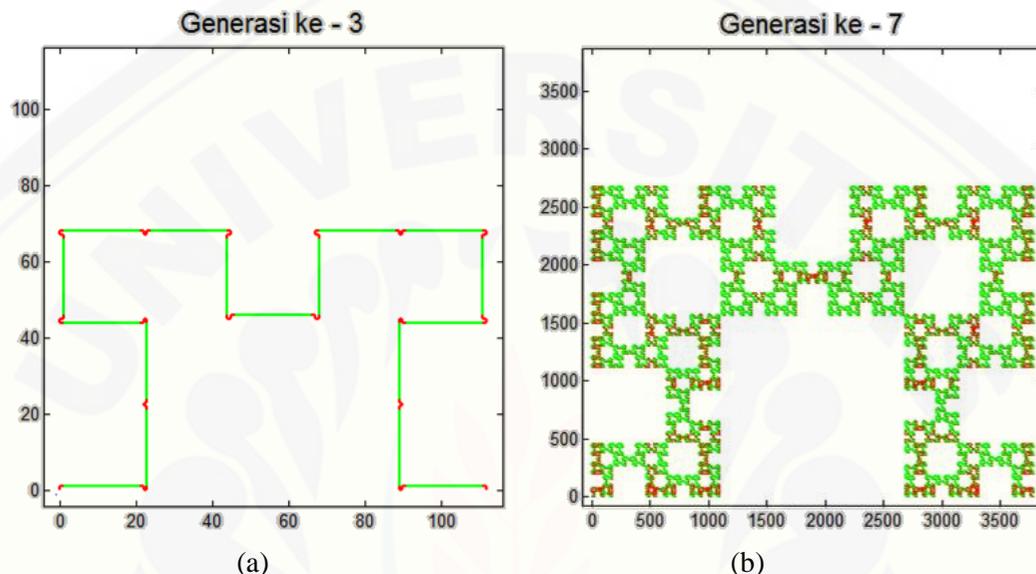


Gambar 4.9 Visualisasi Fraktal *Fibonacci Word* dengan Variasi Nilai *String* K dan $Q = l$.
(a) Generasi ke-3 dan (b) Generasi ke-7

Pada Gambar 4.9 dapat dilihat hasil visualisasi fraktal *Fibonacci word* dengan panjang *string* $L = l$ satuan panjang, $R = l$ satuan panjang, $K = l$ satuan panjang, dan $Q = l$ satuan panjang untuk $l = 1$. Berdasarkan Gambar 4.9, generasi ke-3 dan generasi ke-7 secara keseluruhan memiliki pola yang sama dengan fraktal *Fibonacci word* yang belum divariasikan.

f. Visualisasi fraktal *Fibonacci word* dengan variasi nilai K dan $Q = 20l$

Hasil visualisasi fraktal *Fibonacci word* dengan panjang *string* $L = l$ satuan panjang, $R = l$ satuan panjang, $K = l$ satuan panjang, dan $Q = 20l$ satuan panjang untuk $l = 1$ dapat dilihat pada Gambar 4.10.



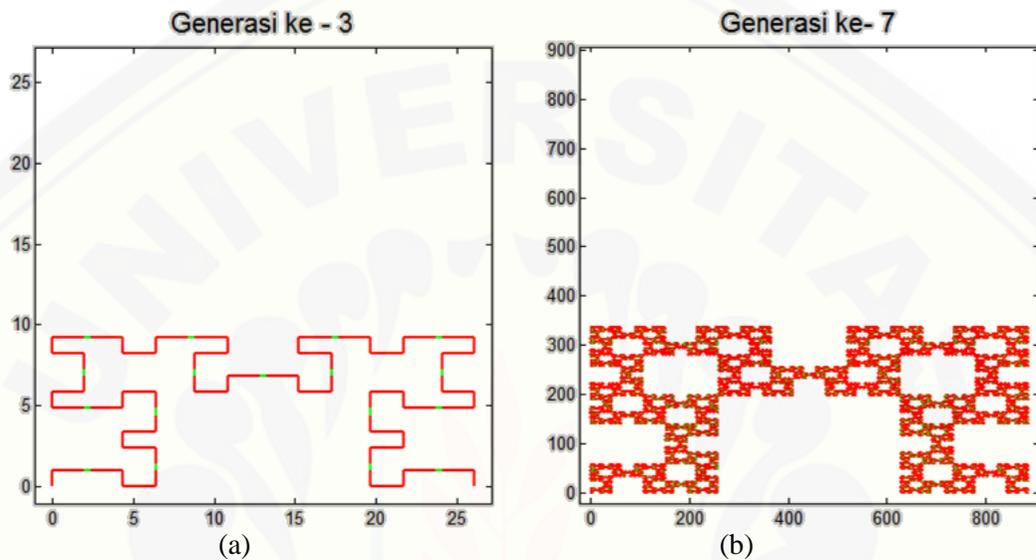
Gambar 4.10 Visualisasi Fraktal *Fibonacci Word* dengan Variasi Nilai *String* K dan $Q = 20l$. (a) Generasi ke-3 dan (b) Generasi ke-7

Hasil pada Gambar 4.10 menunjukkan bahwa generasi ke-3 dan generasi ke-7 secara keseluruhan memiliki pola yang sama dengan fraktal *Fibonacci word* yang belum divariasikan.

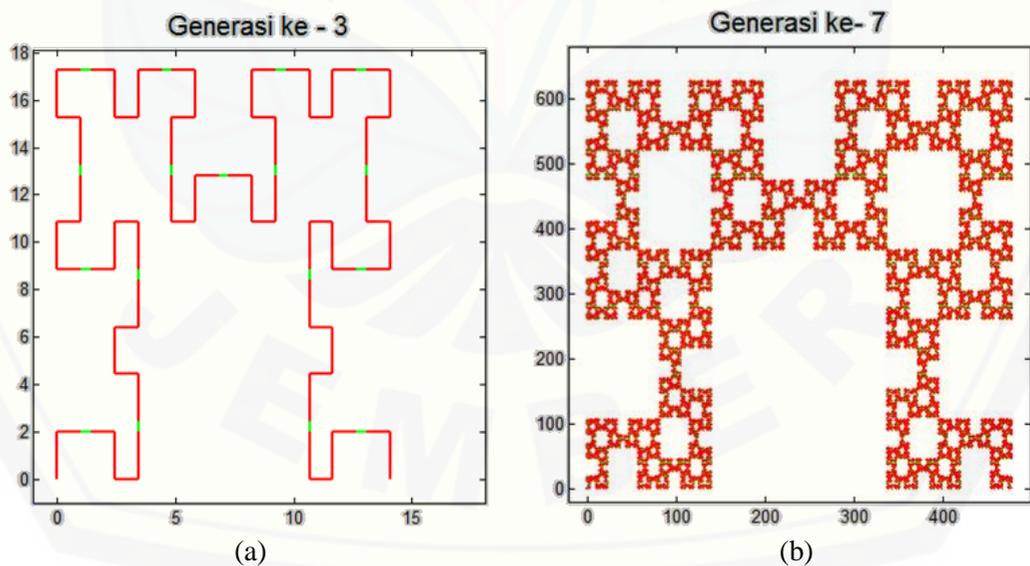
Dari semua hasil variasi panjang segmen *string* K dan $Q = -l$ nilai $-2l$, $-l$, 0 , $\frac{l}{4}$, l dan $20l$, secara keseluruhan diperoleh bentuk pola fraktal yang sama dengan fraktal *Fibonacci word* yang belum divariasikan. Pada Gambar 4.6 dengan nilai panjang segmen *string* K dan $Q = -l$ secara keseluruhan memiliki pola yang sama, namun secara matematis membentuk aturan produksi baru. Sehingga fraktal pada Gambar 4.6 memiliki dua aturan produksi.

Berdasarkan pola fraktal yang diperoleh dari hasil variasi pada program, penulis kembali memvariasikan nilai *string* L dan R , *string* K dan Q , atau *string* L ,

R , K dan Q masing-masing dengan panjang segmen yang berbeda. Hasil visualisasi variasi nilai *string* L dan R dengan panjang segmen yang berbeda dapat dilihat pada Gambar 4.11 dan Gambar 4.12, sedangkan untuk hasil visualisasi variasi lainnya dapat dilihat pada lampiran A.



Gambar 4.11 Visualisasi Fraktal dengan Variasi Nilai *String* $L = 2l$, $R = l$, $K = Q = \frac{l}{1+\sqrt{2}}$, untuk $l = 1$. (a) Generasi ke-3 dan (b) Generasi ke-7.



Gambar 4.12 Visualisasi Fraktal dengan Variasi Nilai *String* $L = l$, $R = 2l$, $K = Q = \frac{l}{1+\sqrt{2}}$, untuk $l = 1$. (a) Generasi ke-3 dan (b) Generasi ke-7.

Secara keseluruhan hasil visualisasi dari semua variasi bentuk fraktal yang diperoleh memiliki pola yang sama dengan fraktal *Fibonacci word* yang belum divariasikan. Sehingga dapat disimpulkan bahwa jika *string* L , R , K dan Q divariasikan dengan nilai yang berbeda tidak mempengaruhi pola fraktal yang dihasilkan, tetapi variasi tersebut mempengaruhi bentuk fraktal yang dihasilkan. Misalnya, jika panjang *string* L atau K yang divariasikan dengan nilai berbeda maka akan diperoleh bentuk fraktal yang lebih panjang atau lebih pendek ke arah horizontal. Sedangkan jika panjang *string* R atau Q yang divariasikan dengan nilai berbeda maka akan diperoleh bentuk fraktal yang lebih panjang atau lebih pendek ke arah vertikal. Untuk visualisasi hasil dari variasi ini dapat dilihat pada lampiran A.

BAB 5. PENUTUP

5.1 Kesimpulan

Berdasarkan dari hasil dan pembahasan pada bab sebelumnya dapat ditarik kesimpulan sebagai berikut.

- a. Bentuk fraktal *Fibonacci word* dari aturan garis ganjil-genap dapat dibangun dengan metode *DOL-systems* dalam dimensi dua yang menggunakan aturan penulisan kembali garis.
- b. Visualisasi program fraktal *Fibonacci word* dengan metode *DOL-systems* yang telah dibuat, memperoleh hasil generasi dan visualisasi bentuk fraktal *Fibonacci word* yang sama dengan penafsiran fraktal *Fibonacci word* dengan metode *DOL-systems* secara matematis dan secara grafis.
- c. Berdasarkan hasil visualisasi variasi panjang segmen dari *string K* dan *Q* dengan nilai $-2l$, $-l$, 0 , $\frac{l}{4}$, l dan $20l$, secara keseluruhan bentuk fraktal yang diperoleh memiliki pola yang sama dengan fraktal *Fibonacci word* yang belum divariasikan. Sehingga dapat disimpulkan bahwa jika *string L*, *R*, *K* dan *Q* divariasikan dengan nilai yang berbeda tidak mempengaruhi pola fraktal yang dihasilkan, tetapi variasi tersebut mempengaruhi bentuk fraktal yang dihasilkan.

5.2 Saran

Konstruksi fraktal *Fibonacci word* masih dapat dikembangkan lagi menggunakan metode-metode lainnya, misalnya konstruksi melibatkan *golden rasio* dan konstruksi menggunakan definisi pemetaan yang berbeda dari barisan *Fibonacci word*.

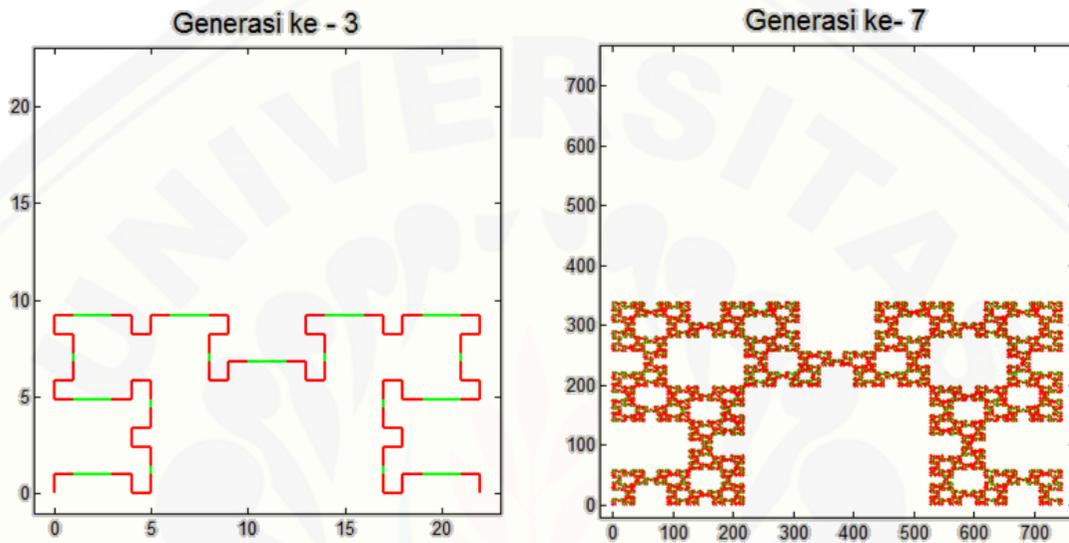
DAFTAR PUSTAKA

- Buorke, P. 2002. *Self Similarity*. <http://paulbourke.net/fractals/fracdim/.htm> [14 Juli 2014].
- Cherbit, G. 1990. *Fractals: Non-integral Dimensions and Applications*. England: Wiley and Sons Ltd.
- Dumaine, A. M. 2009. *The Fibonacci Word Fractal*. https://hal.archivesouvertes.fr/file/index/docid/367972/filename/The_Fibonacci_word_fractal.pdf. [03 Desember 2014].
- Falconer, K. 1990. *Fractal Geometry (Mathematical Foundation and Applications)*. New York: John Wiley and Sons Ltd.
- Ginanjari, W. 2010. *Penerapan L-Systems dalam Membangun Kurva Hilbert Dimensi Dua*. Skripsi. Jember: Universitas Jember.
- Mandelbrot, B. B. 1977. *The Fractal Geometry of Nature*. New York: W. H. Freeman and Company.
- Mardhotillah, T. H. 2012. *Membangun Space-Filling Curve (SFC) dengan Kurva Paeno Menggunakan Pendekatan L-Systems*. Skripsi. Jember: Universitas Jember.
- Nopiyanto, I. 2006. *Membangun Objek-objek Fraktal dengan L-Systems*. Skripsi. Jember: Universitas Jember,
- Ochoa, G. 1998. *An Introduction to Lindenmayer Systems*. http://nsdl.loncapa.org/res/msu/botonl/b_online/e28_3/lsys.html. [23 Desember 2014].
- Peitgen, H.O. dan Soupe, D. 1988. *The Science of Fractal Images*. New York: Springer-Verlag.
- Prusinkiewicz, P. & Lindenmayer, A. 2004. *The Algorithmic Beauty of Plant*. New York: Springer-Verlag.
- Purcell, E. J. dan Varberg, D. 1987. *Kalkulus dan Geometri Analitis Jilid 2*. Edisi V. Diterjemahkan oleh Susila, I. N., Kartasasmita, B., dan Rawuh. Jakarta: Erlangga.
- Wright, D. J. *Dynamical Systems and Fractals Lecture*. <http://www.math.okstate.edu/mathdept/dynamics/lecnotes/lecnotes.html>. [20 Desember 2014].

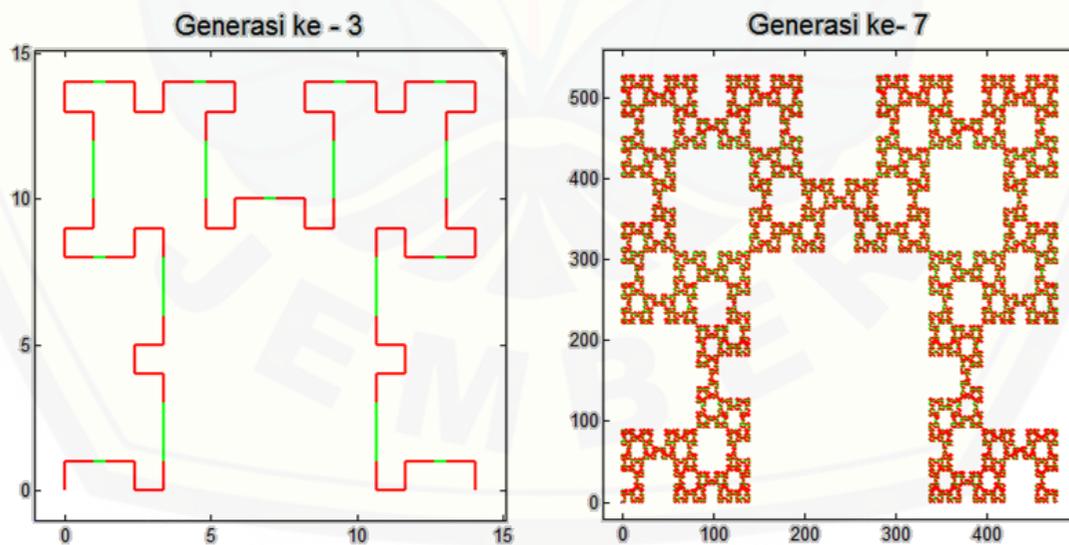
LAMPIRAN

A. Gambar

1. Visualisasi fraktal generasi ke-3 dan generasi ke-7 dari variasi nilai *string* $L = R = l, K = 2l, Q = \frac{l}{1+\sqrt{2}}$, untuk $l = 1$



2. Visualisasi fraktal generasi ke-3 dan generasi ke-7 dari variasi nilai *string* $L = R = l, K = \frac{l}{1+\sqrt{2}}, Q = 2l$, untuk $l = 1$



LKL-R+L+RQR+L-R-LKL-R+L+R-LKL-R-L+RQR+L-R-LKL-R+L+RQR+L-R-L+RQR+L+R-
 LKL-R-L+RQR+L-R-LKL-R+L+R-LKL-R-L+RQR+L+R-LKL-R+L+RQR+L-R-LKL-R+L+R-
 LKL-R-L+RQR+L-R-LKL-R+L+RQR+L-R-L+RQR+L+R-LKL-R+L+RQR+L-R-LKL-R+L+R-
 LKL-R-L+RQR+L+R-LKL-R+L+RQR+L-R-L+RQR+L+R-LKL-R-L+RQR+L-R-LKL-R+L+R-
 LKL-R-L+RQR+L+R-LKL-R+L+RQR+L-R-LKL-R+L+R-LKL-R-L+RQR+L-R-LKL-
 R+L+RQR+L-R-L+RQR+L+R-LKL-R-L+RQR+L-R-LKL-R+L+R-LKL-R-L+RQR+L+R-LKL-
 R+L+RQR+L-R-L+RQR+L+R-LKL-R-L+RQR+L-R-LKL-R+L+RQR+L-R-L+RQR+L+R-LKL-
 R+L+RQR+L-R-LKL-R+L+R-LKL-R-L+RQR+L

C. Script Program

1. Menu utama

```

clc; clear all; close all;
tanda0='';
tanda1='';
all_g0=['L'
        'R'];
all_g1=['+R-LKL-R'
        '-L+RQR+L'];
matrik_g1=[];
g0='L';
awal=g0(end);
iter=7;% iterasi
K=-1;
Q=-1;
R1=1;
L1=1;
A=[0;0] ;%x,y
B=A;
%=====
for i=1:iter
    if i==1
        if g0=='L'
            g1=all_g1(1,:);
        elseif g0=='R'
            g1=all_g1(2,:);
        end
    else
        for j=1:2
            if j==1
                tanda0=g1(end-1);
                tanda1=g1(end);
                g3=g1;
                % ganti tanda +/-
                g3=ganti_tanda(tanda0,g3);
                %ganti huruf R/L/Q/K
                g3=ganti_huruf(tanda1,g3);
            end
        end
    end
end

```

```
g042=g3;
%-----
n1=length(g1);
g01=g1;

tanda0=g3(end-1);
tanda1=g3(end);
% ganti tanda +/-
g01=ganti_tanda(tanda0,g01);
%ganti huruf R/L/Q/K
g01=ganti_huruf(tanda1,g01);
g041=g01;

g01(1)='-'; % sambungan 1
if awal=='R'
g01(1)='+';
end
g3(n1+1:n1+n1)=g01;
%-----
g042(1)='-'; % sambungan 4
g041(1)='+'; % sambungan 3
if awal=='R'
g042(1)='+';
g041(1)='-';
end
g4=[g041 g042];
%-----
n1=length(g3);
n2=length(g0);

tanda0=g3(end-1);
tanda1=g3(end);
% ganti tanda +/-
g01=ganti_tanda(tanda0,g0);
%ganti huruf R/L/Q/K
g01=ganti_huruf(tanda1,g0);

g01(1)='+'; % sambungan 2
if awal=='R'
g01(1)='-';
end
g3(n1+1:n1+n2)=g01;
else
n1=length(g3);
n2=length(g4);

g3(n1+1:n1+n2)=g4;
end
end
```

```

if i>1
    g0='';
    g0=g1;
end
    g1=g3;
end
if g0=='L'
    g0='+L';
elseif g0=='R'
    g0='+R';
end
disp(['generasi ke' num2str(i)]);
disp(g1);
disp(' ');
end

matrik_g1=matrik(g1);
n=length(matrik_g1);

A1=A; B1=B;
s=0;
for i=1:n % panjang LRQK
    c=0;
    while c==0
        s=s+1;
        if g1(s)=='L'
            p=L1;
            c=1;
        elseif g1(s)=='R'
            p=R1;
            c=1;
        elseif g1(s)=='K'
            u=K;
            c=0;
        elseif g1(s)=='Q'
            u=Q;
            c=0;
        end
    end
end

=====
    B=B+matrik_g1(:,i)*p;
if max(matrik_g1(:,i))==2
    [a b]=max(matrik_g1(:,i));
    if b==1
        plot([A(1) A(1)+u],[A(2) A(2)],'g','linewidth',2);
        A=A+[u 0]';
        B=B-[1*p-u 0]';
    else
        plot([A(1) A(1)],[A(2) A(2)+u],'g','linewidth',2);

```

```

        A=A+[0 u]';
        B=B-[0 1*p-u]';
    end
elseif min(matrik_g1(:,i))==-2
    [a b]=min(matrik_g1(:,i));
    if b==1
        plot([A(1) A(1)-u],[A(2) A(2)],'g','linewidth',2);
        A=A-[u 0]';
        B=B+[1*p-u 0]';
    else
        plot([A(1) A(1)], [A(2) A(2)-u], 'g', 'linewidth', 2);
        A=A-[0 u]';
        B=B+[0 1*p-u]';
    end
end
plot([A(1) B(1)], [A(2) B(2)], 'r', [-1 -1], [-1 -1], 'linewidth', 2);
hold on
A=B;
A1=A;
B1=B;
% pause(1)
end

axis square
title(['Generasi ke- ' num2str(iter)]);
if awal=='L'
xlim([-1 B(1)+1]);
ylim([-1 B(1)+1]);
end

```

2. Script untuk mengubah huruf

```

function g3=ganti_huruf(tanda1,g3)

    if tanda1=='L' && g3(2)=='L'
        indx=findstr('L',g3);
        for ii=1:length(indx)
            g3(indx(ii))='r';
        end
        indx=findstr('R',g3);
        for ii=1:length(indx)
            g3(indx(ii))='L';
        end
    elseif tanda1=='R' && g3(2)=='R'
        indx=findstr('L',g3);
        for ii=1:length(indx)
            g3(indx(ii))='r';
        end
    end

```

```

    indx=findstr('R',g3);
    for ii=1:length(indx)
        g3(indx(ii))='L';
    end
end
g3=upper(g3);
indx=findstr('RKR',g3);
for ii=1:length(indx)
    g3(indx(ii):indx(ii)+2)='RqR';
end
indx=findstr('LQL',g3);
for ii=1:length(indx)
    g3(indx(ii):indx(ii)+2)='LKL';
end
g3=upper(g3);

```

3. Script untuk mengubah tanda

```

function g3=ganti_tanda(tanda0,g3)
    if tanda0=='-' && g3(1)=='+'
        indx=findstr('-',g3);
        for ii=1:length(indx)
            g3(indx(ii))='*';
        end

        indx=findstr('+',g3);
        for ii=1:length(indx)
            g3(indx(ii))='-';
        end
        indx=findstr('*',g3);
        for ii=1:length(indx)
            g3(indx(ii))='+';
        end
    elseif tanda0=='+' && g3(1)=='-'
        indx=findstr('-',g3);
        for ii=1:length(indx)
            g3(indx(ii))='*';
        end

        indx=findstr('+',g3);
        for ii=1:length(indx)
            g3(indx(ii))='-';
        end
        indx=findstr('*',g3);
        for ii=1:length(indx)
            g3(indx(ii))='+';
        end
    end
end

```

4. Script untuk menggambar

```
function [A]=matrik(g1)
n=length(g1);
%arah awal utara atau timur
% utara=1; barat=2; selatan=3; timur=4;
A=[] ;%x,y
if g1(1)=='+'
    arah=1; % utara
    A(:,1)=[0 1]';
else
    arah=4; %timur
    A(:,1)=[1 0]';
end
i1=1;
for i=3:2:n
    i1=i1+1;
    % asal arah
    if arah==1 % utara
        if g1(i)=='+' % artinya ke kiri atau ke barat
            A(:,i1)=[-1 0]'; % x,y
            arah=2;
        elseif g1(i)=='-' % artinya ke kanan atau ke timur
            A(:,i1)=[1 0]'; %x,y
            arah=4;
        else % artinya lurus atau ke utara
            A(:,i1)=[0 2]'; %x,y
            arah=1;
        end
    elseif arah==2 % barat
        if g1(i)=='+' % artinya ke kiri atau ke selatan
            A(:,i1)=[0 -1]'; % x,y
            arah=3;
        elseif g1(i)=='-' % artinya ke kanan atau ke utara
            A(:,i1)=[0 1]'; %x,y
            arah=1;
        else % artinya lurus atau ke barat
            A(:,i1)=[-2 0]'; %x,y
            arah=2;
        end
    elseif arah==3 % selatan
        if g1(i)=='+' % artinya ke kiri atau ke timur
            A(:,i1)=[1 0]'; % x,y
            arah=4;
        elseif g1(i)=='-' % artinya ke kanan atau ke barat
            A(:,i1)=[-1 0]'; %x,y
            arah=2;
```

```
else % artinya lurus atau ke selatan
    A(:,i1)=[0 -2]'; %x,y
    arah=3;
end

elseif arah==4 % timur
    if g1(i)=='+' % artinya ke kiri atau ke utara
        A(:,i1)=[0 1]'; % x,y
        arah=1;
    elseif g1(i)=='-' % artinya ke kanan atau ke selatan
        A(:,i1)=[0 -1]'; %x,y
        arah=3;
    else % artinya lurus atau ke timur
        A(:,i1)=[2 0]'; %x,y
        arah=4;
    end
end
end
```