



**PEMANFAATAN FRAKTAL *FIBONACCI SNOWFLAKE*  
PADA POLA PENGUBINAN**

**SKRIPSI**

Oleh

**Farah Intan Nur Oktavia  
NIM 151810101039**

**JURUSAN MATEMATIKA  
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
UNIVERSITAS JEMBER  
2019**



**PEMANFAATAN FRAKTAL *FIBONACCI SNOWFLAKE*  
PADA POLA PENGUBINAN**

**SKRIPSI**

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

Oleh

**Farah Intan Nur Oktavia  
NIM 151810101039**

**JURUSAN MATEMATIKA  
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
UNIVERSITAS JEMBER  
2019**

## PERSEMBAHAN

Skripsi ini saya persembahkan untuk:

1. Ibunda Lindatus Salamah dan Ayahanda Subaidi tercinta, yang telah membesarkan, mendidik dan mendukung dengan penuh kasih sayang;
2. Adik Qotrunnada Nur Jihan Adila tersayang yang selalu mendoakan dan memberikan dukungan dan semangat;
3. Guru-guru sejak taman kanak-kanak sampai dengan perguruan tinggi;
4. Keluarga besar yang selalu mendoakan dan memberikan dukungan untuk menyelesaikan pendidikan;
5. Almamater Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember.

**MOTTO**

**Man Shobaro Dzofiro**

“Barang siapa yang bersabar maka akan beruntung.”\*)

“Sesungguhnya sesudah kesulitan itu ada kemudahan.”

( Terjemahan Al-Qur’an Surah Al-Insyirah ayat 6)\*\*)



---

\*) Fuadi, A. 2009. *Negeri 5 Menara*. Jakarta: PT. Gramedia.

\*\*\*) Departemen Agama Republik Indonesia. 2012. *Al-Qur'an Cordoba Special for Muslimah*. Bandung: PT. Cordoba Internasional.

**PERNYATAAN**

Saya yang bertanda tangan di bawah ini:

nama : Farah Intan Nur Oktavia

NIM : 151810101039

menyatakan dengan sesungguhnya bahwa skripsi yang berjudul “Pemanfaatan Fraktal *Fibonacci Snowflake* Pada Pola Pengubinan” 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 di kemudian hari pernyataan ini tidak benar.

Jember, 22 Januari 2019

Yang menyatakan,

Farah Intan Nur Oktavia

NIM 151810101039

**SKRIPSI**

**PEMANFAATAN FRAKTAL *FIBONACCI SNOWFLAKE*  
PADA POLA PENGUBINAN**

Oleh

Farah Intan Nur Oktavia  
NIM 151810101039

Pembimbing;

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

Dosen Pembimbing Anggota : Dr. Firdaus Ubaidillah, S.Si., M.Si.

**PENGESAHAN**

Skripsi berjudul “Pemanfaatan Fraktal *Fibonacci Snowflake* Pada Pola Pengubinan” telah diuji dan disahkan pada:

hari, tanggal :

tempat : Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas  
Jember

Tim Penguji:

Ketua,

Anggota I,

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

Dr. Firdaus Ubaidillah, S.Si., M.Si.  
NIP. 197006061998031003

Anggota II,

Anggota III,

Bagus Juliyanto, S.Si., M.Si.  
NIP. 198007022003121001

Dr. Kristiana Wijaya, S.Si., M.Si.  
NIP. 197408132000032004

Mengesahkan  
Dekan,

Drs. Sujito, Ph.D  
NIP. 196102041987111001

## RINGKASAN

**Pemanfaatan Fraktal *Fibonacci Snowflake* Pada Pola Pengubinan;** Farah Intan Nur Oktavia, 151810101039; 2019: 38 halaman; Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember.

Fraktal *Fibonacci Snowflake* merupakan pengembangan dari fraktal *Fibonacci Word* yang memiliki struktur seperti *snowflake*. Fraktal *Fibonacci Snowflake* adalah sebuah polyomino yang menata bidang atau permukaan dengan translasi. Polyomino berarti suatu bidang atau daerah yang terbagi oleh banyak persegi.

A. Blondin-Masse (2012) mendefinisikan *Fibonacci Snowflake* adalah kurva tertutup sederhana pada kisi persegi  $Z^2$  terkait dengan deret *Fibonacci*  $F_n$ . *Fibonacci Snowflake* ini termasuk kelas kurva yang panjangnya dinyatakan dalam rumus  $4F_{3n+1}$  dan bagian dalam atau interior kurva yang membentuk ubin pada bidang melalui translasi. Fraktal *Fibonacci Snowflake* dapat dipresentasikan oleh  $\phi_n = (q_{3n+1})^3 q_{3n+1}$  dengan  $q_n$  adalah barisan *fibonacci* yang didefinisikan secara rekursif sebagai:  $q_n = q_{n-1}q_{n-2}$  jika  $n \equiv 2 \pmod{3}$  dan  $q_n = q_{n-1}\overline{q_{n-2}}$  jika  $n \equiv 0,1 \pmod{3}$ , untuk  $n \geq 2$ ;  $q_0 = \varepsilon$  (kosong) dan  $q_1 = R$ .

A. Blondin-Masse (2011) menyebutkan bahwa fraktal *Fibonacci Snowflake* dapat dimanfaatkan dalam pola pengubinan. O' Daffer (2008) mendefinisikan pengubinan adalah suatu pola khusus yang terdiri dari bangun - bangun geometri yang disusun tanpa pemisah/jarak untuk menutupi suatu bidang datar.

Penelitian yang akan dilakukan ini mengkaji tentang pembangkitan fraktal *Fibonacci Snowflake* yang dapat dimanfaatkan dalam pola pengubinan. Berdasarkan penelitian yang telah dilakukan diperoleh bahwa, hasil kurva fraktal *Fibonacci Snowflake* dapat diterapkan pada pola pengubinan dengan translasi karena kurva fraktal *Fibonacci Snowflake* memenuhi syarat pola pengubinan, yaitu saling menutupi tanpa ada tumpang-tindih. Translasi dilakukan dengan menggandakan kurva fraktal *Fibonacci Snowflake* namun posisinya digeser menggunakan arah kanan atas, kanan bawah, kiri atas dan kiri bawah.



Selain hal tersebut di atas, juga dapat disimpulkan bahwa nilai generasi berdampak pada banyaknya segmen garis yang dihasilkan. Semakin besar generasi suatu fraktal, maka semakin banyak pula segmen garis yang dihasilkan. Hal itu disebabkan karena  $n$  yang lebih besar mempunyai lebih banyak komponen penyusunnya.



## PRAKATA

Puji syukur penulis panjatkan kehadiran Allah SWT, yang telah melimpahkan rahmat dan hidayah-Nya sehingga penulis dapat menyelesaikan skripsi yang berjudul “Pemanfaatan Fraktal *Fibonacci Snowflake* Pada Pola Pengubinan”. 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. Kosala Dwidja Purnomo, S.Si., M.Si., selaku Dosen Pembimbing Utama dan Dr. Firdaus Ubaidillah, S.Si., M.Si., selaku Dosen Pembimbing Anggota yang dengan penuh kesabaran membimbing, mengarahkan, memberikan saran dan petunjuk dalam penyusunan skripsi ini;
2. Bagus Juliyanto, S.Si., M.Si. dan Dr. Kristiana Wijaya, S.Si., M.Si., selaku Dosen Penguji yang telah memberikan kritik dan saran yang membangun dalam penyusunan skripsi;
3. Ziaul Arif, S.Si., M.Si., dan Dr. Kristiana Wijaya, S.Si., M.Si., selaku Dosen Pembimbing Akademik yang telah membimbing, mengarahkan, memberikan saran dan petunjuk dalam masa perkuliahan;
4. Seluruh staf pengajar Jurusan Matematika Fakultas MIPA Universitas Jember yang telah memberikan ilmu serta bimbingannya sehingga penulis dapat menyelesaikan skripsi ini;
5. Ibu Lindatus Salamah, Bapak Subaidi, Adik Ading, Nenek Umi dan seluruh keluarga di rumah yang telah memberikan doa dan motivasi;
6. Sahabatku IS, Rika, Nadiya, Eriska, Mela, Mitha dan Rivi yang telah membantu, memberikan kritik & saran, memberikan motivasi, mendoakan dan memberikan semangat;
7. Sahabat SIGMA '15 yang telah menemani selama masa perkuliahan;

8. Teman-teman seperjuangan bidang minat fraktal, diantaranya Kakroz, Mela, Nadiya, Ingka, Novita, Ipin, Ijak dan Mitha yang selalu kompak;
9. Teman-teman condro *squad*, Tika, Diah, Firda, Zahro dan Fira yang selalu memotivasi dan mendoakan yang terbaik untukku;
10. Teman-teman GenBI yang telah memberi banyak pengalaman dan dukungan;
11. 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, Januari 2019

Penulis

DAFTAR ISI

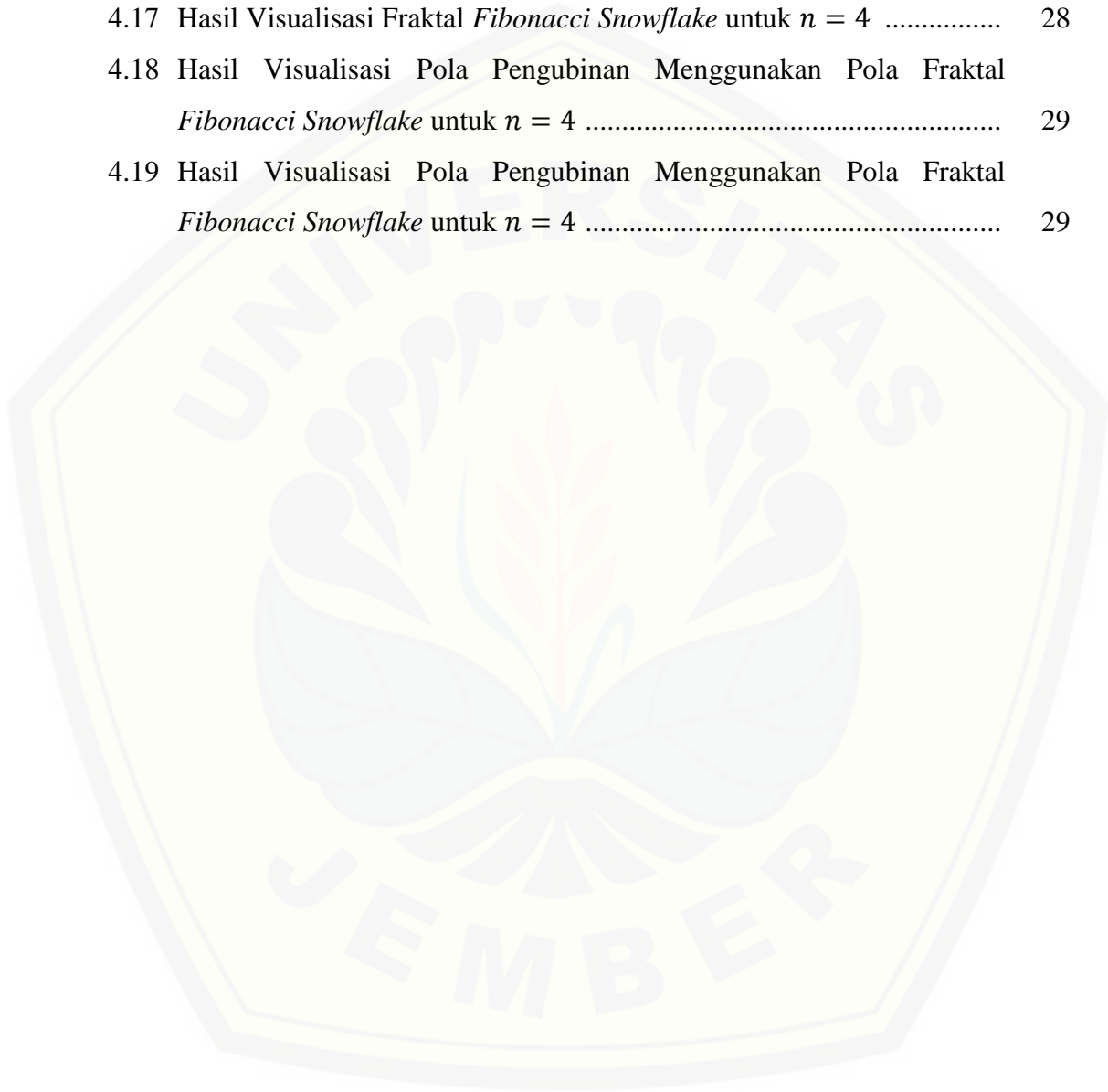
	Halaman
HALAMAN JUDUL .....	i
HALAMAN PERSEMBAHAN .....	ii
HALAMAN MOTTO .....	iii
HALAMAN PERNYATAAN .....	iv
HALAMAN PEMBIMBING .....	v
HALAMAN PENGESAHAN .....	vi
RINGKASAN .....	vii
PRAKATA .....	ix
DAFTAR ISI.....	xi
DAFTAR GAMBAR .....	xiii
DAFTAR TABEL .....	xv
DAFTAR LAMPIRAN .....	xvi
<b>BAB 1. PENDAHULUAN .....</b>	<b>1</b>
<b>1.1 Latar Belakang Masalah .....</b>	<b>1</b>
<b>1.2 Rumusan Masalah .....</b>	<b>2</b>
<b>1.3 Tujuan Masalah .....</b>	<b>2</b>
<b>1.4 Manfaat .....</b>	<b>3</b>
<b>BAB 2. TINJAUAN PUSTAKA .....</b>	<b>4</b>
<b>2.1 Fraktal .....</b>	<b>4</b>
<b>2.2 Barisan <i>Fibonacci Word</i> .....</b>	<b>4</b>
<b>2.3 Fraktal <i>Fibonacci Word</i>.....</b>	<b>5</b>
<b>2.4 Fraktal <i>Fibonacci Snowflake</i>.....</b>	<b>6</b>
<b>2.5 Aturan Konstruksi Fraktal <i>Fibonacci Snowflake</i> .....</b>	<b>8</b>
<b>2.6 Pengubinan (<i>Tesselation</i>).....</b>	<b>9</b>
<b>BAB 3. METODE PENELITIAN .....</b>	<b>12</b>
<b>3.1 Studi Literatur .....</b>	<b>12</b>
<b>3.2 Penafsiran Fraktal <i>Fibonacci Snowflake</i> secara Matematis. ....</b>	<b>13</b>
<b>3.3 Penafsiran Fraktal <i>Fibonacci Snowflake</i> secara Grafis .....</b>	<b>13</b>

3.4 Pembuatan Program .....	13
3.5 Analisis Hasil .....	14
<b>BAB 4. HASIL DAN PEMBAHASAN .....</b>	<b>15</b>
4.1 Penafsiran Fraktal <i>Fibonacci Snowflake</i> secara Matematis.	15
4.2 Penafsiran Fraktal <i>Fibonacci Snowflake</i> secara Grafis .....	16
4.3 Pembuatan Program .....	19
4.4 Hasil Simulasi Program .....	20
4.5 Pembahasan .....	22
<b>BAB 5. PENUTUP .....</b>	<b>32</b>
5.1 Kesimpulan .....	32
5.2 Saran .....	32
<b>DAFTAR PUSTAKA .....</b>	<b>33</b>
<b>LAMPIRAN .....</b>	<b>34</b>

DAFTAR GAMBAR

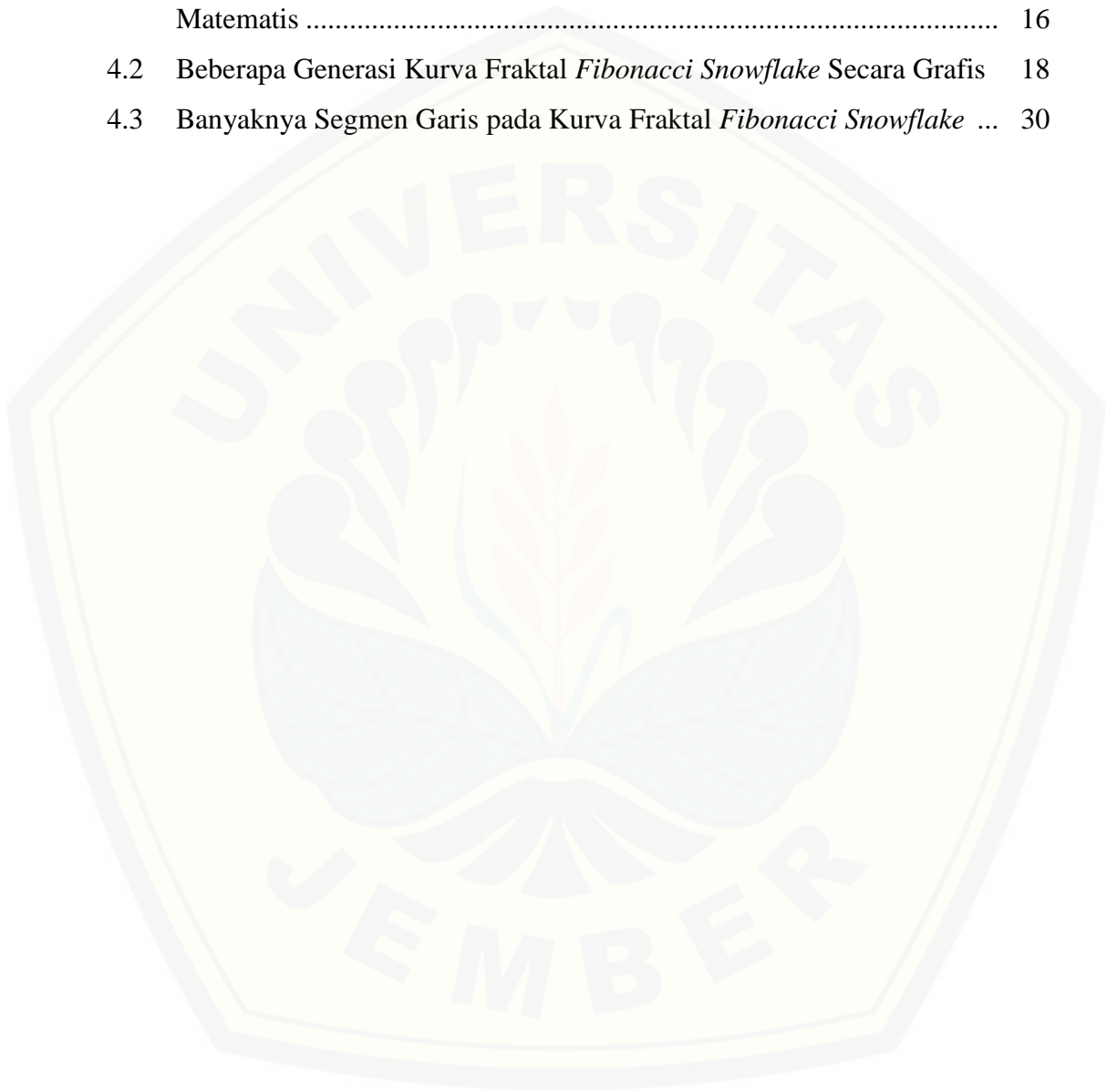
	Halaman
2.1 Contoh Fraktal .....	4
2.2 Konstruksi Fraktal <i>Fibonacci Word</i> .....	6
2.3 Fraktal <i>Fibonacci Snowflake</i> $\phi_n$ untuk $n = 0, 1, 2, 3, 4$ .....	7
2.4 Pengubinan ( <i>Tesselation</i> ) dengan Fraktal <i>Fibonacci Snowflake</i> .....	8
2.5 <i>Regular Tesselation</i> .....	9
2.6 <i>Semiregular Tesselation</i> .....	10
2.7 <i>Non-regular Tesselation</i> .....	10
3.1 Skema Metode Penelitian .....	12
4.1 Hasil Visualisasi Fraktal <i>Fibonacci Snowflake</i> untuk Generasi $n = 0$ .	16
4.2 Hasil Visualisasi Fraktal <i>Fibonacci Snowflake</i> untuk Generasi $n = 1$ .	17
4.3 Hasil Visualisasi Fraktal <i>Fibonacci Snowflake</i> .....	21
4.4 Hasil Visualisasi Fraktal <i>Fibonacci Snowflake</i> .....	21
4.5 Hasil Visualisasi Fraktal <i>Fibonacci Snowflake</i> untuk $n = 0$ .....	22
4.6 Hasil Visualisasi Pola Pengubinan Menggunakan Pola Fraktal <i>Fibonacci Snowflake</i> untuk $n = 0$ .....	23
4.7 Hasil Visualisasi Pola Pengubinan Menggunakan Pola Fraktal <i>Fibonacci Snowflake</i> untuk $n = 0$ .....	23
4.8 Hasil Visualisasi Fraktal <i>Fibonacci Snowflake</i> untuk $n = 1$ .....	24
4.9 Hasil Visualisasi Pola Pengubinan Menggunakan Pola Fraktal <i>Fibonacci Snowflake</i> untuk $n = 1$ .....	24
4.10 Hasil Visualisasi Pola Pengubinan Menggunakan Pola Fraktal <i>Fibonacci Snowflake</i> untuk $n = 1$ .....	25
4.11 Hasil Visualisasi Fraktal <i>Fibonacci Snowflake</i> untuk $n = 2$ .....	25
4.12 Hasil Visualisasi Pola Pengubinan Menggunakan Pola Fraktal <i>Fibonacci Snowflake</i> untuk $n = 2$ .....	26
4.13 Hasil Visualisasi Pola Pengubinan Menggunakan Pola Fraktal <i>Fibonacci Snowflake</i> untuk $n = 2$ .....	26
4.14 Hasil Visualisasi Fraktal <i>Fibonacci Snowflake</i> untuk $n = 3$ .....	27

4.15 Hasil Visualisasi Pola Pengubinan Menggunakan Pola Fraktal <i>Fibonacci Snowflake</i> untuk $n = 3$ .....	27
4.16 Hasil Visualisasi Pola Pengubinan Menggunakan Pola Fraktal <i>Fibonacci Snowflake</i> untuk $n = 3$ .....	28
4.17 Hasil Visualisasi Fraktal <i>Fibonacci Snowflake</i> untuk $n = 4$ .....	28
4.18 Hasil Visualisasi Pola Pengubinan Menggunakan Pola Fraktal <i>Fibonacci Snowflake</i> untuk $n = 4$ .....	29
4.19 Hasil Visualisasi Pola Pengubinan Menggunakan Pola Fraktal <i>Fibonacci Snowflake</i> untuk $n = 4$ .....	29



**DAFTAR TABEL**

	Halaman
4.1 Beberapa Generasi Kurva Fraktal <i>Fibonacci Snowflake</i> Secara Matematis .....	16
4.2 Beberapa Generasi Kurva Fraktal <i>Fibonacci Snowflake</i> Secara Grafis	18
4.3 Banyaknya Segmen Garis pada Kurva Fraktal <i>Fibonacci Snowflake</i> ...	30





**DAFTAR LAMPIRAN**

	Halaman
Lampiran 4.1 <i>Script</i> Program .....	34



## BAB 1. PENDAHULUAN

### 1.1 Latar Belakang

Matematika merupakan ilmu pengetahuan universal yang mendasari perkembangan teknologi modern. Matematika berkembang pesat seiring dengan perkembangan teknologi, terutama setelah ditemukannya komputer. Salah satu bidang matematika yang berkembang pesat adalah geometri fraktal. Keberadaan geometri fraktal menunjukkan bahwa matematika bukanlah ilmu yang datar, tetapi merupakan ilmu yang indah yang dapat menghasilkan karya-karya yang memiliki nilai seni tinggi. Karena keindahannya, geometri fraktal banyak digunakan dalam *computer graphics* untuk menciptakan bentuk-bentuk yang alami dan menakjubkan.

Menurut Peitgen (1998), 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, sedangkan fraktal merupakan kelas bentuk geometri kompleks yang umumnya mempunyai nilai dimensi pecahan. Fraktal sendiri diperkenalkan pertama kali oleh Benoit B Mandelbrot, seorang matematikawan dari Polandia.

Mandelbrot (1983) membagi fraktal menjadi dua jenis. Fraktal jenis pertama adalah himpunan-himpunan fraktal (*fractal sets*), contohnya *Koch snowflake*, *Sierpinski triangle*, *Mandelbrot set*, dan *Julia set*. Sedangkan fraktal jenis kedua adalah fraktal alami (*natural fractal*), contohnya cabang-cabang pohon, bentuk pegunungan, garis pantai, dan daun.

Salah satu bentuk fraktal jenis pertama adalah kurva *Fibonacci word*. Kurva ini dikenalkan pertama kali oleh Dumaine (2009) yang kemudian dikenal sebagai fraktal *Fibonacci Word*. Fraktal *Fibonacci Word* adalah suatu kurva yang memiliki sifat *self-similarity* melalui aturan gambar yang sederhana dan menarik berdasarkan pada barisan *Fibonacci Word*. Terdapat pengembangan dari Fraktal *Fibonacci Word*, yaitu fraktal *Fibonacci Snowflake*.

Fraktal *Fibonacci Snowflake* adalah sebuah polyomino yang menata bidang atau permukaan dengan translasi. Polyomino sendiri berasal dari kata “*poly*” yang

berarti banyak dan “*domino*” yang berarti suatu daerah yang terbagi oleh dua persegi. Sehingga polyomino berarti suatu bidang atau daerah yang terbagi oleh banyak persegi. Fraktal ini mempunyai struktur seperti *snowflake*, yaitu dibentuk dengan membuat penambahan secara terus menerus bentuk yang sama. A. Blondin-Masse (2011) menyebutkan bahwa fraktal *Fibonacci Snowflake* dapat dimanfaatkan dalam pola pengubinan (*tesselation*).

Tesselasi (*tesselation*) adalah suatu pengulangan pola dari bangun-bangun yang menutup secara lengkap suatu bidang datar tanpa ada celah atau tumpang tindih. Tesselasi juga biasa disebut pengubinan. Tesselasi merupakan konsep antar cabang ilmu pengetahuan, yakni matematika dan seni. Dalam bidang seni, tesselasi mengacu pada konsep artistik. Sedangkan dalam matematika, tesselasi meliputi beberapa konsep-konsep matematika yang lebih dalam seperti segi banyak beraturan, segi banyak tidak beraturan, kekongruenan, sudut dalam, jumlah sudut dalam suatu segi banyak, simetri, translasi, refleksi, dan rotasi (Rokhmah, 2010).

Berdasarkan uraian di atas, peneliti tertarik untuk mengkaji lebih lanjut pembangkitan kurva fraktal berdasarkan aturan konstruksi fraktal *Fibonacci Snowflake*. Selain itu, peneliti akan mengkaji pola pengubinan (*tesselation*) yang dimiliki oleh fraktal *Fibonacci Snowflake*.

## 1.2 Rumusan Masalah

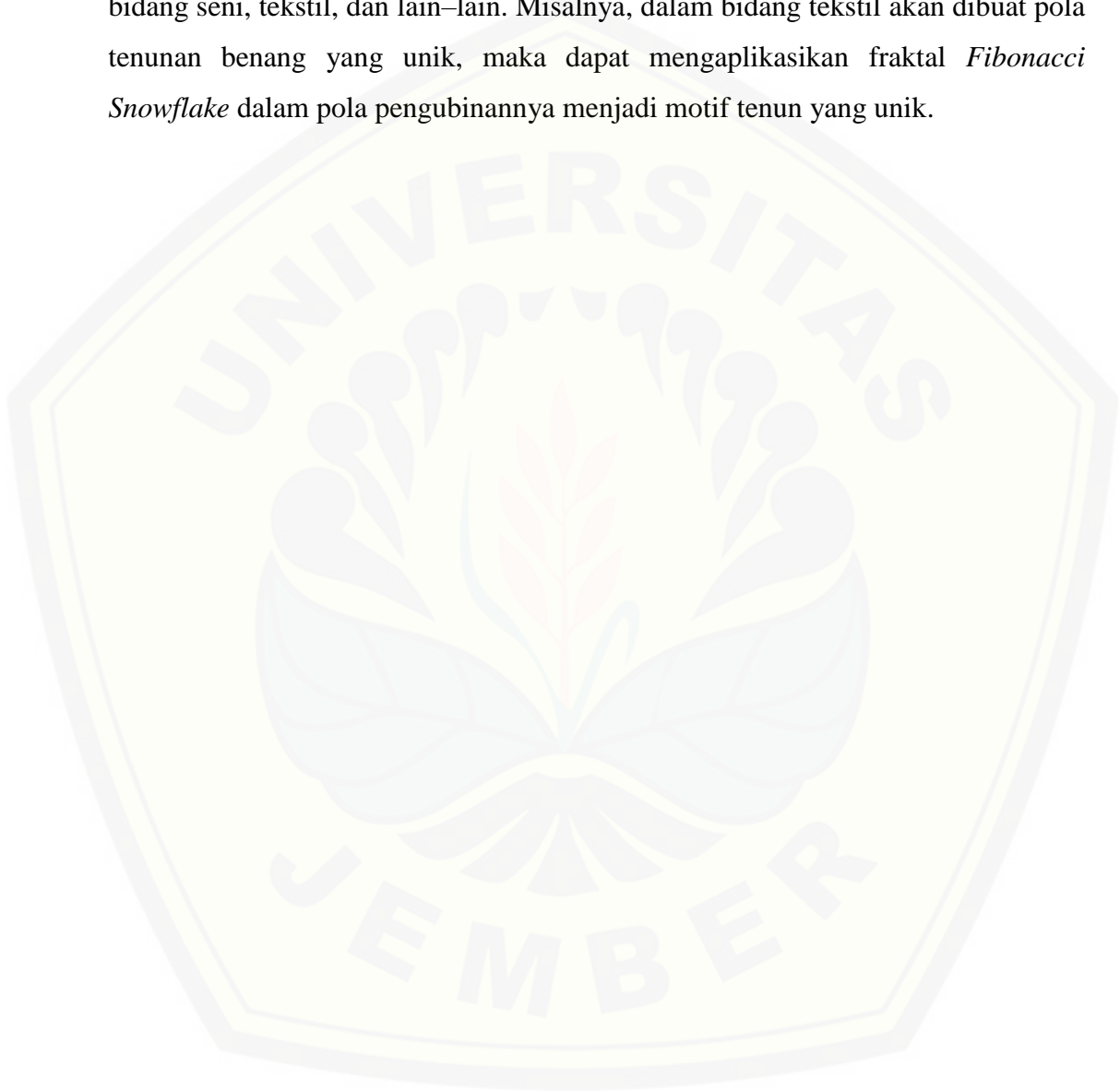
Berdasarkan uraian latar belakang tersebut, permasalahan yang akan dibahas dalam penelitian ini adalah bagaimana mengkonstruksi fraktal *Fibonacci Snowflake* yang dapat dimanfaatkan pada pola pengubinan.

## 1.3 Tujuan

Tujuan dari penelitian ini adalah mengetahui konstruksi fraktal *Fibonacci Snowflake* yang dapat dimanfaatkan pada pola pengubinan.

#### 1.4 Manfaat

Manfaat dari penelitian ini adalah untuk mengetahui dan memanfaatkan bentuk fraktal *Fibonacci Snowflake* pada pengubinan (*tesselation*). Hasil dari penelitian ini dapat diaplikasikan dalam kehidupan sehari-hari, contohnya dalam bidang seni, tekstil, dan lain-lain. Misalnya, dalam bidang tekstil akan dibuat pola tenunan benang yang unik, maka dapat mengaplikasikan fraktal *Fibonacci Snowflake* dalam pola pengubinannya menjadi motif tenun yang unik.

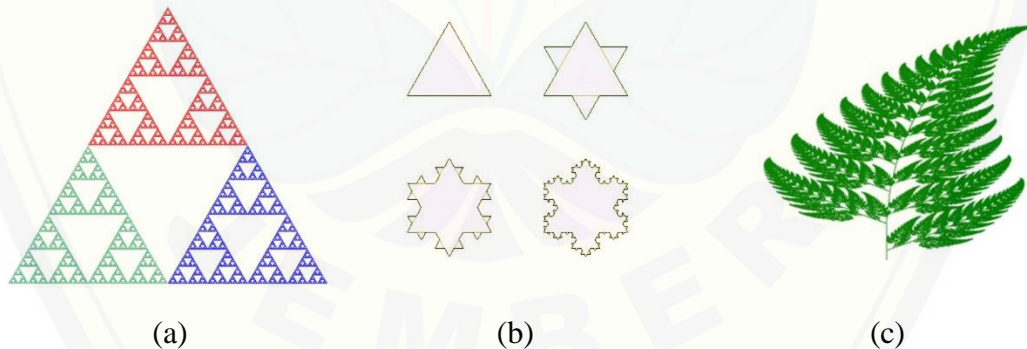


## BAB 2. TINJAUAN PUSTAKA

### 2.1 Fraktal

Istilah fraktal pertama kali diperkenalkan oleh matematikawan kelahiran Polandia bernama Benoit B. Mandelbrot. Menurut Mandelbrot (1983), fraktal berasal dari bahasa latin yaitu *frangere* merupakan kata kerja yang berarti membelah atau *fractus* merupakan kata sifat yang berarti tidak teratur atau terfragmentasi.

Fraktal mempunyai dua ciri khas, yaitu *self-similarity* dan *infinite detail*. *Self similarity* merupakan keadaan objek yang dibangun secara berulang dengan mengganti suatu gambar dengan yang sebangun, tetapi berukuran lebih kecil dari aslinya. Artinya setiap bagian kecil dalam sebuah fraktal dapat dipandang sebagai replikasi skala kecil dari bentuk keseluruhan. *Infinite detail* merupakan objek fraktal yang memiliki bentuk besar yang seakan-akan tidak habis-habis apabila diperhatikan. Contohnya kurva koch apabila diperbesar dengan generasi yang tak terhingga akan mempunyai ketidakteraturan yang sama (Santosa, 1994). Beberapa contoh fraktal dapat dilihat pada Gambar 2.1.



(a) *Sierpinski Triangle*, (b) *Koch Snowflake*, (c) Fraktal Alami Bentuk Daun

Gambar 2.1 Contoh Fraktal  
(Sumber : [www.wikiwand.com](http://www.wikiwand.com))

### 2.2 Barisan *Fibonacci Word*

Tahun 1202, Leonardo Fibonacci memperkenalkan sebuah barisan yang dikenal sebagai barisan *Fibonacci*. Barisan ini didefinisikan secara rekursif sebagai:

$$F_n = F_{n-1} + F_{n-2}$$

dengan  $F_1 = 1$  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 (0 – 1). Barisan ini didefinisikan secara induktif sebagai:

$$f_1 = 1$$

$$f_2 = 0$$

$$f_n = f_{n-1}f_{n-2}, n \geq 3$$

Fungsi  $f_n$  merupakan gabungan dari dua sifat sebelumnya. Berikut adalah barisan *Fibonacci Word* berturut-turut:

$$f_1 = 1$$

$$f_2 = 0$$

$$f_3 = 01$$

$$f_4 = 010$$

$$f_5 = 01001$$

$$f_6 = 01001010$$

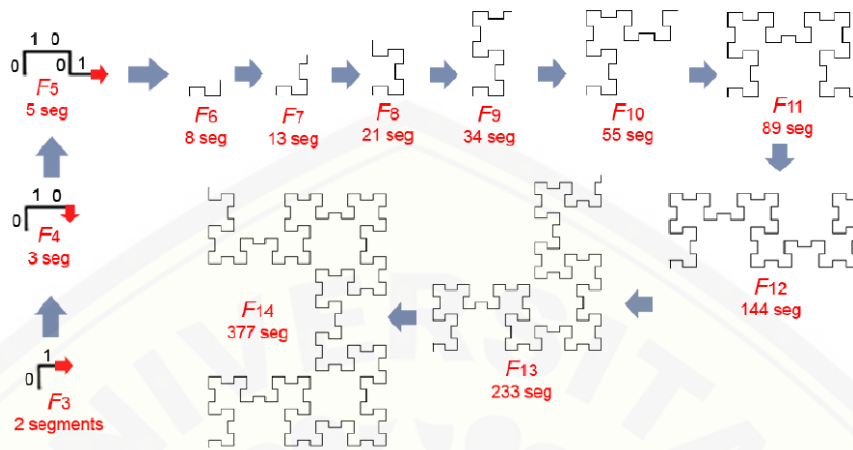
$$f_7 = 0100101001001$$

### 2.3 Fraktal *Fibonacci Word*

Fraktal *Fibonacci Word* adalah suatu kurva yang memiliki sifat *self-similarity* melalui aturan gambar yang sederhana dan menarik berdasarkan pada barisan *Fibonacci Word*. Aturan konstruksi fraktal *Fibonacci Word* dari simbol barisan *Fibonacci Word* dapat dilakukan dengan menggunakan aturan garis ganjil-genap. Secara umum aturan tersebut adalah:

- a. Apabila digitnya adalah 0, maka gambarkan segmen garis dan setelahnya berbelok ke kanan atau ke kiri, yaitu berbelok ke kanan jika digit 0 terletak pada suku ganjil dan berbelok ke kiri jika digit 0 terletak pada suku genap,
- b. Apabila digitnya adalah 1, maka gambarkan segmen garis dan tidak berbelok setelahnya (Dumaine, 2009).

Konstruksi fraktal *Fibonacci word* menggunakan aturan garis ganjil – genap dapat dilihat pada Gambar 2.2



Gambar 2.2 Konstruksi Fraktal *Fibonacci Word* (Dumaine, 2009)

### 2.4 Fraktal *Fibonacci Snowflake*

Fraktal *Fibonacci Snowflake* merupakan pengembangan dari fraktal *Fibonacci Word* yang memiliki struktur seperti *snowflake*. *Fibonacci Snowflake* adalah kurva tertutup sederhana pada kisi persegi  $Z^2$  terkait dengan deret *Fibonacci*  $F_n$ . *Fibonacci Snowflake* ini termasuk kelas kurva yang panjangnya dinyatakan dalam rumus  $4F_{3n+1}$  dan bagian dalam atau interior kurva yang membentuk ubin pada bidang melalui translasi (A. Blondin-Masse, 2012).

Pada tahun sebelumnya, A. Blondin-Masse (2011) mendefinisikan *Fibonacci Snowflake* atau ubin pada orde  $n$  merupakan sebuah polyomino  $T_F(n)$  yang dipresentasikan oleh:

$$\phi_n = (q_{3n+1})^3 q_{3n+1} \tag{2.1}$$

dengan:

$\phi_n$  adalah kurva fraktal *Fibonacci Snowflake* ke- $n$ ,

$q$  adalah barisan *fibonacci* yang didefinisikan secara rekursif dan  $n \geq 0$ .

Berikut adalah beberapa generasi untuk fraktal *Fibonacci Snowflake*:

$$\phi_0 = (q_1)^3 q_1 = (q_1)^4$$

$$\phi_1 = (q_4)^4$$

$$\phi_2 = (q_7)^4$$

$$\phi_3 = (q_{10})^4$$

$$\phi_4 = (q_{13})^4$$

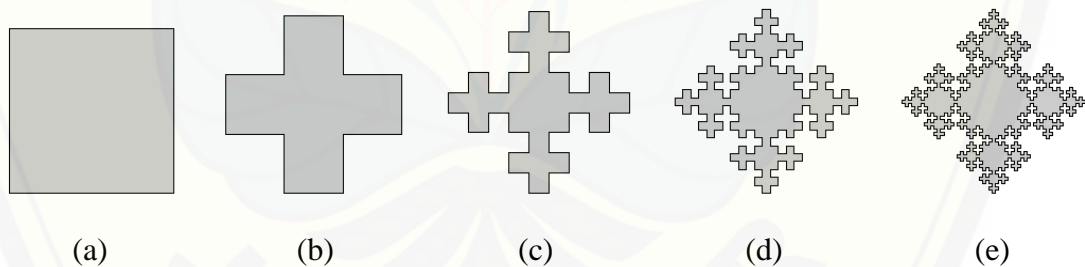
$$\phi_5 = (q_{16})^4$$

Perhatikan bahwa tidak semua nilai  $q_n$  dapat dipresentasikan oleh  $\phi_n$ . Hanya  $q_1, q_4, q_7, q_{10}$  dan seterusnya dengan beda 3.

Polyomino yang dimaksud adalah kesatuan unit yang terbatas kisi kotak (piksel) dalam bidang diskrit yang batasnya adalah jalur tertutup sederhana. Dikatakan jalur tertutup sederhana jika titik awal sama dengan titik akhir dengan tidak memotong dirinya sendiri. Secara khusus, sebuah polyomino sederhana terhubung (tanpa lubang), dan batasnya sederhana (tidak menyilang). Fraktal ini termasuk dalam *Fibonacci* karena perimeter yang digunakan adalah aturan angka *Fibonacci* yakni  $4F_{(3n+1)}$  dimana  $F_{(n)}$  adalah barisan *fibonacci* dengan  $F_{(1)} = F_{(2)} = 1$ . Fraktal ini mempunyai struktur seperti *snowflake* yang dinyatakan dalam barisan:

1, 5, 29, 169, 985, 5741, 33461, 195025, 1136689, 6625109, 38613965, ...

Konstruksi fraktal *Fibonacci Snowflake* dapat dilihat pada Gambar 2.3.



(a)

(b)

(c)

(d)

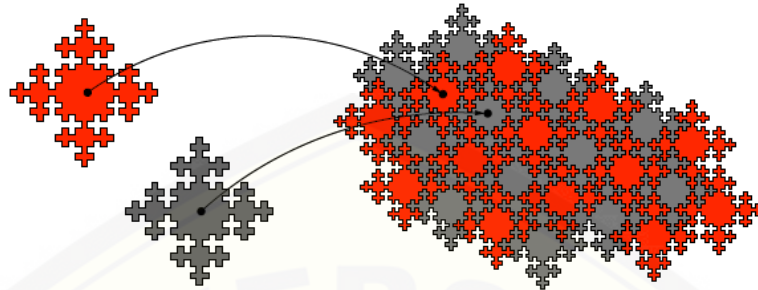
(e)

(a)  $n = 0$ , (b)  $n = 1$ , (c)  $n = 2$ , (d)  $n = 3$ , (e)  $n = 4$

Gambar 2.3 Fraktal *Fibonacci Snowflake*  $\phi_n$  untuk  $n = 0, 1, 2, 3, 4$ .



Pengubinan menggunakan *Fibonacci Snowflake* dapat dilihat pada Gambar 2.4.



Gambar 2.4 Pengubinan (*tesselation*) dengan Fraktal *Fibonacci Snowflake*

### 2.5 Aturan Konstruksi Fraktal *Fibonacci Snowflake*

Untuk mendeskripsikan jalur pada kotak persegi adalah dengan memberikan urutan arah pada titik pertemuan atau titik potong: jalan sepenuhnya ditentukan oleh langkah awal  $\alpha \in \varepsilon$ , dimana  $\varepsilon$  adalah huruf kosong yang berarti diam tidak bergerak. Kemudian urutan indikasi arah, yaitu bergerak ke kiri ( $L$ ), kanan ( $R$ ), maju ( $F$ ) atau mundur ( $B$ ). Dalam aturan konstruksi fraktal *Fibonacci Snowflake* didefinisikan pada alfabet  $\tau = \{L, R\}$ , dan barisan diperoleh dari huruf-huruf yang didefinisikan oleh rekurensi:

$$q_n = \begin{cases} q_{n-1}q_{n-2} & \text{jika } n \equiv 2 \pmod{3}, \\ q_{n-1}\overline{q_{n-2}} & \text{jika } n \equiv 0,1 \pmod{3}, \end{cases} \text{ untuk } n \geq 2. \quad (2.2)$$

dengan:

$$q_0 = \varepsilon \text{ (kosong),}$$

$$q_1 = R,$$

Operasi  $\bar{\cdot}$  berarti lawan atau komplemen dari  $R$  ataupun  $L$ . Misal  $q_1 = R$  maka  $\overline{q_1} = L$ .

Berikut adalah beberapa iterasi untuk fraktal *Fibonacci Snowflake*:

$$q_0 = \varepsilon$$

$$q_1 = R$$

$$q_2 = R$$

$$q_3 = RL$$

$$q_4 = RLL$$

$$q_5 = RLLRL$$

$$q_6 = RLLRLLRR$$

$$q_7 = RLLRLLRRLRRLR$$

$$q_8 = RLLRLLRRLRRLRLLRLLRR$$

Perhatikan bahwa  $|q_n| = F_n$  adalah angka Fibonacci ke- $n$ , dengan  $|q_n|$  adalah jumlah komponen  $q_n$ .

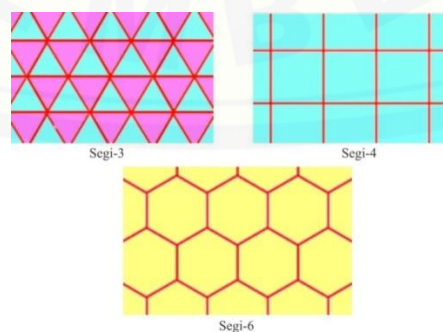
## 2.6 Pengubinan (*Tesselation*)

Dalam matematika pengubinan disebut dengan tessulasi. Kata tessulasi berasal dari bahasa Inggris yaitu *tesselation*. Pengubinan adalah kata benda yang berasal dari kata ubin yakni semacam benda yang digunakan untuk menutup lantai. Suatu pola khusus yang terdiri dari bangun-bangun geometri yang disusun tanpa pemisah/jarak untuk menutupi suatu bidang datar dinamakan pengubinan (O' Daffer, 2008).

Terdapat beberapa macam *tesselation* menurut bidang datar pembangunnya, yakni *regular tesselation*, *semiregular tesselation*, dan *non-regular tesselation*.

### a. *Regular Tesselation*

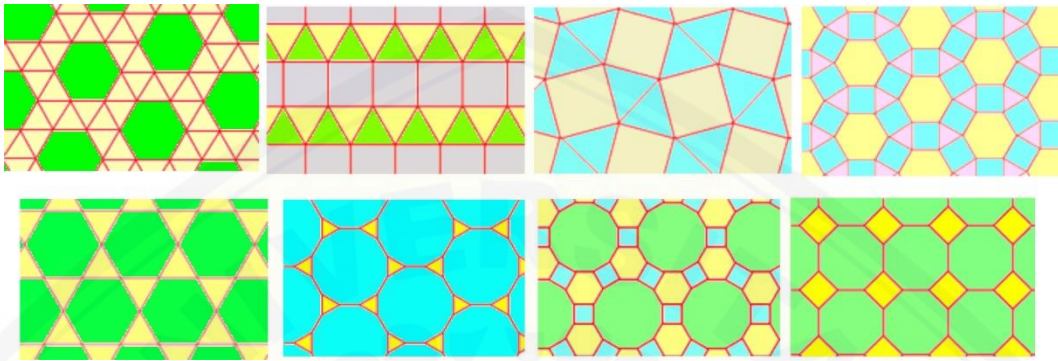
Sebuah *tesselation* yang seluruhnya terbentuk dari bangun segi- $n$  yang beraturan, dengan bangun tersebut saling berhimpitan dinamakan *regular tesselation* (O' Daffer, 2008). Terdapat 3 contoh *regular tesselation*, yakni *tesselation* yang dibentuk oleh segi-3 beraturan (segitiga sama sisi), segi-4 beraturan (persegi), dan segi-6 beraturan (heksagon).



Gambar 2.5 *Regular Tesselation*  
(Sumber : [www.mathisfun.com](http://www.mathisfun.com))

### b. Semiregular Tessellation

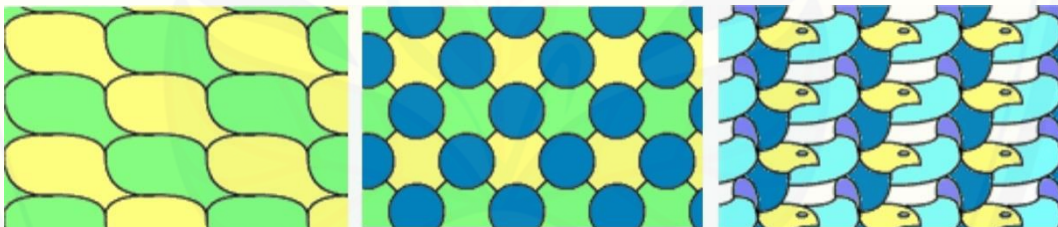
*Semiregular tessellation* merupakan tessellation yang dibentuk dari dua atau lebih bangun segi- $n$  beraturan, dengan pola susunan bangun pada tiap *vertex* harus sama. Berikut beberapa contoh *semiregular tessellation*:



Gambar 2.6 *Semiregular Tessellation*  
(Sumber : [www.mathisfun.com](http://www.mathisfun.com))

### c. Non-regular Tessellation

*Tessellation* yang dibangun menggunakan bangun-bangun datar yang tidak beraturan dinamakan *non-regular tessellation*.



Gambar 2.7 *Non-regular Tessellation*  
(Sumber : [www.mathisfun.com](http://www.mathisfun.com))

Untuk menentukan pengubinan bangun-bangun segi- $n$  beraturan, harus dipahami terlebih dahulu tentang besar setiap sudut pada segi- $n$  beraturan. Misalnya, jumlah ukuran sudut segitiga adalah  $180^\circ$  dan besar ukuran sudut satu lingkaran penuh adalah  $360^\circ$ . Untuk mengetahui segi banyak beraturan yang bisa digunakan untuk pengubinan, akan dicari menggunakan langkah berikut.

- Pertama, besar sudut sebuah segi- $n$  beraturan adalah  $\frac{(n-2)}{n} \cdot 180^\circ$
- Selanjutnya, agar tidak saling tindih dan tidak ada celah, maka  $p$  buah ubin tersebut harus tepat menutup permukaan. Ini berarti,

$$p \cdot \pi \left(1 - \frac{2}{n}\right) = 2\pi$$

$$p \cdot \left(1 - \frac{2}{n}\right) = 2$$

$$1 - \frac{2}{n} = \frac{2}{p}$$

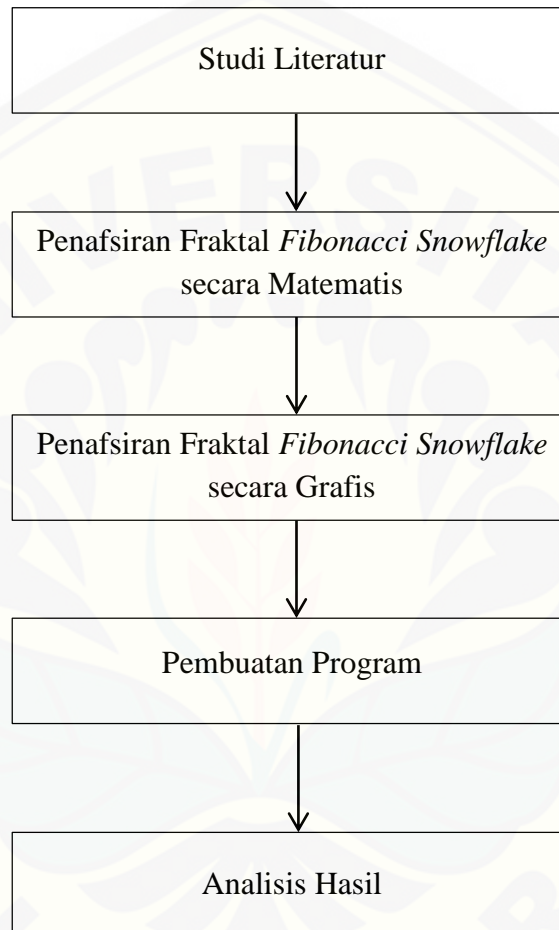
$$\frac{2}{n} + \frac{2}{p} = 1$$

$$\frac{1}{n} + \frac{1}{p} = \frac{1}{2} \tag{2.3}$$

Persamaan (2.3) hanya dipenuhi berturut-turut untuk nilai  $n = 3, p = 6, n = 4, p = 4$  dan  $n = 6, p = 3$ . Dengan demikian, hanya terdapat tepat 3 (tiga) bangun datar yang bisa digunakan pada pengubinan beraturan, yaitu segitiga samasisi, persegi dan segienam beraturan.

### BAB 3. METODE PENELITIAN

Langkah-langkah yang akan dilakukan dalam penelitian ini 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 Studi Literatur

Studi literatur dilakukan dengan mencari dan mempelajari referensi yang berkaitan dengan fraktal *Fibonacci Snowflake* dan permasalahan pengubinan pada jurnal, buku dan skripsi yang berhubungan dengan permasalahan tersebut.

### 3.2 Penafsiran Fraktal *Fibonacci Snowflake* secara Matematis

Langkah pertama dalam penelitian ini adalah penafsiran fraktal *Fibonacci Snowflake* secara matematis. Pada langkah ini akan ditentukan komponen-komponen pembentukan kurva fraktal *Fibonacci Snowflake*, dengan komponen-komponen ini mengacu pada definisi barisan *Fibonacci Snowflake* pada persamaan (2.2). Selain itu, komponen-komponen pembentukan kurva fraktal *Fibonacci Snowflake* juga diperoleh berdasarkan aturan konstruksi kurva yang mengacu pada persamaan (2.1). Kemudian berdasarkan komponen-komponen pembentukan kurva tersebut akan diperoleh beberapa generasi kurva fraktal *Fibonacci Snowflake*.

### 3.3 Penafsiran Fraktal *Fibonacci Snowflake* secara Grafis

Langkah kedua dalam penelitian ini adalah penafsiran fraktal *Fibonacci Snowflake* secara grafis setelah didapatkan penafsiran secara matematisnya pada langkah pertama. Hasil generasi kurva fraktal *Fibonacci Snowflake* pada langkah sebelumnya, kemudian digambar secara grafis sesuai komponen-komponen pembentukan kurva yang telah didapatkan sebelumnya. Pada langkah ini akan digambarkan kurva fraktal *Fibonacci Snowflake* hingga beberapa generasi sebagai pembandingan dengan hasil visualisasi pada program. Kemudian akan divisualisasikan pula pola pengubinan dengan memanfaatkan fraktal *Fibonacci Snowflake*.

### 3.4 Pembuatan Program

Langkah ketiga dalam penelitian ini yaitu pembuatan program visualisasi berdasarkan kurva yang dihasilkan di langkah kedua. Algoritma program yang digunakan dalam membangkitkan kurva adalah sebagai berikut:

- a. Mendefinisikan simbol untuk menggambar;
- b. Menentukan aturan konstruksi fraktal *Fibonacci Snowflake* dalam *Matlab*;
- c. Menentukan nilai generasi fraktal *Fibonacci Snowflake* yang akan divisualisasi sebagai input;

- d. Menentukan panjang segmen, sudut putar, arah, posisi titik awal dan aturan pola pengubinan pada fraktal *Fibonacci Snowflake*;
- e. Mengiterasikan nilai generasi hingga generasi ke- $n$  berdasarkan aturan konstruksi yang telah diberikan;

### 3.5 Analisis Hasil

Hasil yang diperoleh dari pembuatan program adalah visualisasi kurva dari fraktal *Fibonacci Snowflake* dalam dimensi dua dan visualisasi pola pengubinan dengan menggunakan kurva dari fraktal *Fibonacci Snowflake*. Dari kurva yang dihasilkan akan dianalisa mengenai jenis *tesselation* dari hasil kurva fraktal *Fibonacci Snowflake*.

## BAB 5. PENUTUP

### 5.1 Kesimpulan

Berdasarkan dari hasil dan pembahasan, diperoleh kesimpulan bahwa hasil kurva fraktal *Fibonacci Snowflake* dapat diterapkan pada pola pengubinan dengan translasi karena kurva fraktal *Fibonacci Snowflake* memenuhi syarat pola pengubinan, yaitu saling menutupi tanpa ada tumpang-tindih. Translasi dilakukan dengan menggandakan kurva fraktal *Fibonacci Snowflake* namun posisinya digeser menggunakan arah kanan atas, kanan bawah, kiri atas dan kiri bawah.

### 5.2 Saran

Pola pengubinan menggunakan fraktal *Fibonacci Snowflake* dapat dikembangkan lagi terutama untuk aturan konstruksi fraktal *Fibonacci Snowflake* yang berbeda. Sehingga dimungkinkan untuk dapat menghasilkan variasi pola pengubinan menggunakan fraktal *Fibonacci Snowflake*.



DAFTAR PUSTAKA

- Beng, P. D. R. 2018. *Tessellation*. <https://www.mathisfun.com/geometry/tessellation.html>. [Diakses pada 12 Mei 2018]
- Dumaine, A. M. 2009. *The Fibonacci Word Fractal*. [https://hal.archives-ouvertes.fr/file/index/docid/367972/filename/The\\_Fibonacci\\_word\\_fractal.pdf](https://hal.archives-ouvertes.fr/file/index/docid/367972/filename/The_Fibonacci_word_fractal.pdf). [Diakses pada 12 Mei 2018]
- Kontributor Wikipedia. 2018. *Fraktal*. [Http://www.wikiwand.com/id/Fraktal](http://www.wikiwand.com/id/Fraktal). [Diakses pada 12 Mei 2018]
- Mandelbrot, B. B. 1983. *The Fractal Geometry of Nature*. New York: W. H. Freeman and Company.
- Massé A. B., S. Brlek, A. Garon, S. Labbé. 2011. Two Infinite Families of Polyominoes That Tile the Plane by Translation In Two Distinct Ways. *Theoret. Comput. Sci.* 412 4778–4786.
- Massé A. B., S. Brlek, S.Labbé, M. M. France. 2011. Fibonacci Snowflakes. *Ann. Sci. Math. Québec* 35, No 2, 141–152.
- Massé A. B., S. Brlek, S.Labbé, M. M. France. 2012. Complexity of the Fibonacci Snowflake. *Fractals*.20 257–260.
- O’Daffer, P. G. 2008. *Mathematics for Elementary School Teachers*. Fourth Edition: Pearson Education.
- Peitgen, H.O. dan D. Soupe. 1988. *The Science of Fractal Images*. New York: Springer-Verlag.
- Rokhmah, S. 2010. *Empowering Student’s Creativity Through Learning Tessellation Using the Internet*. Tersedia pada [http://ifed.or.id/v2/index.php?option=com\\_content&view=article&id=147:empowering-students-creativity-through-learning-tesselation-using-theinternet&catid=45:pendidikan&Itemid=54&lang=in](http://ifed.or.id/v2/index.php?option=com_content&view=article&id=147:empowering-students-creativity-through-learning-tesselation-using-theinternet&catid=45:pendidikan&Itemid=54&lang=in). [Diakses pada 19 Mei 2018]
- Santosa, P. I. 1994. *Grafika Komputer dan Antarmuka Grafis Teknik Penyusunan Program Aplikasi Berbasis Grafis yang Profesional*. Yogyakarta : Andi Offset.

## LAMPIRAN

### Lampiran A *Script* Program

```

function varargout = GuiFibSnowflake(varargin)
% GUIFIBSNOWFLAKE MATLAB code for GuiFibSnowflake.fig
%   GUIFIBSNOWFLAKE, by itself, creates a new GUIFIBSNOWFLAKE
or raises the existing
%   singleton*.
%
%   H = GUIFIBSNOWFLAKE returns the handle to a new
GUIFIBSNOWFLAKE or the handle to
%   the existing singleton*.
%
%   GUIFIBSNOWFLAKE('CALLBACK',hObject,eventData,handles,...)
calls the local
%   function named CALLBACK in GUIFIBSNOWFLAKE.M with the given
input arguments.
%
%   GUIFIBSNOWFLAKE('Property','Value',...) creates a new
GUIFIBSNOWFLAKE or raises the
%   existing singleton*. Starting from the left, property
value pairs are
%   applied to the GUI before GuiFibSnowflake_OpeningFcn gets
called. An
%   unrecognized property name or invalid value makes property
application
%   stop. All inputs are passed to GuiFibSnowflake_OpeningFcn
via varargin.
%
%   *See GUI Options on GUIDE's Tools menu. Choose "GUI allows
only one
%   instance to run (singleton)".
%
% See also: GUIDE, GUIDATA, GUIHANDLES

% Edit the above text to modify the response to help
GuiFibSnowflake

% Last Modified by GUIDE v2.5 16-Nov-2018 10:28:07

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',  gui_Singleton, ...
                  'gui_OpeningFcn', @GuiFibSnowflake_OpeningFcn,
                  ...
                  'gui_OutputFcn',  @GuiFibSnowflake_OutputFcn,
                  ...
                  'gui_LayoutFcn',  [] , ...
                  'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});

```

```
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT

% --- Executes just before GuiFibSnowflake is made visible.
function GuiFibSnowflake_OpeningFcn(hObject, eventdata, handles,
varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to GuiFibSnowflake (see
VARARGIN)

% Choose default command line output for GuiFibSnowflake
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);
clc;
movegui(gcf, 'center');
set(handles.edit1, 'string', '');
set(handles.popupmenu1, 'value', 1);
cla(handles.axes1, 'reset');
axes(handles.axes1);
axis off
set(handles.uitable1, 'data', []);
% UIWAIT makes GuiFibSnowflake wait for user response (see
UIRESUME)
% uiwait(handles.figure1);

% --- Outputs from this function are returned to the command line.
function varargout = GuiFibSnowflake_OutputFcn(hObject, eventdata,
handles)
% varargout  cell array for returning output args (see VARARGOUT);
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure
varargout{1} = handles.output;

function edit1_Callback(hObject, eventdata, handles)
% hObject    handle to edit1 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject, 'String') returns contents of edit1 as text
```

```

%         str2double(get(hObject,'String')) returns contents of
edit1 as a double

% --- Executes during object creation, after setting all
properties.
function edit1_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit1 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    empty - handles not created until after all
CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%         See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on button press in pushbutton1.
function pushbutton1_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton1 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)
clc;
cla(handles.axes1,'reset');
axes(handles.axes1);
axis off
set(handles.uitable1,'data',[]);
pause(0.1);
n=str2num(get(handles.edit1,'string'));
c=get(handles.popupmenu1,'value');
if c==1
    color=[.3 .3 .3];
elseif c==10
    color=rand(1,3);
else
    fixed='rgbcmykw';
    color=fixed(c-1);
end
[xcoor,ycoor]=Coordinate(n);
[xcoor2,ycoor2]=Coordinate(n-1);
axes(handles.axes1);
patch(xcoor,ycoor,color);
axis image
axis off
set(handles.uitable1,'data',[xcoor' ycoor'],'userdata',[xcoor2'
ycoor2']);

% --- Executes on button press in pushbutton2.
function pushbutton2_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton2 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

```

```

clc;
coor=get(handles.uitable1,'data');
coor2=get(handles.uitable1,'userdata');
c=get(handles.popupmenu1,'value');
if c==1
    color=[.3 .3 .3];
elseif c==10
    color=rand(1,3);
else
    fixed='rgbcmykw';
    color=fixed(c-1);
end
if ~isempty(coor)
    xcoor=coor(:,1);
    ycoor=coor(:,2);
    L1=max(xcoor)-min(xcoor);
    if ~isempty(coor2)
        xcoor2=coor2(:,1);
        L2=max(xcoor2)-min(xcoor2);
    else
        L2=0;
    end
    xcoor=xcoor+ceil(L2/2);
    ycoor=ycoor+ceil(L1/2);
    patch(xcoor,ycoor,color);
    set(handles.uitable1,'data',[xcoor ycoor]);
end

% --- Executes on button press in pushbutton3.
function pushbutton3_Callback(hObject, eventdata, handles)
% hObject      handle to pushbutton3 (see GCBO)
% eventdata    reserved - to be defined in a future version of
MATLAB
% handles      structure with handles and user data (see GUIDATA)
clc;
coor=get(handles.uitable1,'data');
coor2=get(handles.uitable1,'userdata');
c=get(handles.popupmenu1,'value');
if c==1
    color=[.3 .3 .3];
elseif c==10
    color=rand(1,3);
else
    fixed='rgbcmykw';
    color=fixed(c-1);
end
if ~isempty(coor)
    xcoor=coor(:,1);
    ycoor=coor(:,2);
    L1=max(xcoor)-min(xcoor);
    if ~isempty(coor2)
        xcoor2=coor2(:,1);
        L2=max(xcoor2)-min(xcoor2);
    else
        L2=0;
    end
    xcoor=xcoor+ceil(L1/2);
    ycoor=ycoor-ceil(L2/2);
    patch(xcoor,ycoor,color);

```

```

        set(handles.uitable1, 'data', [xcoor ycoor]);
end

% --- Executes on button press in pushbutton4.
function pushbutton4_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton4 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)
clc;
coor=get(handles.uitable1, 'data');
coor2=get(handles.uitable1, 'userdata');
c=get(handles.popupmenu1, 'value');
if c==1
    color=[.3 .3 .3];
elseif c==10
    color=rand(1,3);
else
    fixed='rgbcmykw';
    color=fixed(c-1);
end
if ~isempty(coor)
    xcoor=coor(:,1);
    ycoor=coor(:,2);
    L1=max(xcoor)-min(xcoor);
    if ~isempty(coor2)
        xcoor2=coor2(:,1);
        L2=max(xcoor2)-min(xcoor2);
    else
        L2=0;
    end
    xcoor=xcoor-ceil(L1/2);
    ycoor=ycoor+ceil(L2/2);
    patch(xcoor,ycoor,color);
    set(handles.uitable1, 'data', [xcoor ycoor]);
end

% --- Executes on button press in pushbutton5.
function pushbutton5_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton5 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)
clc;
coor=get(handles.uitable1, 'data');
coor2=get(handles.uitable1, 'userdata');
c=get(handles.popupmenu1, 'value');
if c==1
    color=[.3 .3 .3];
elseif c==10
    color=rand(1,3);
else
    fixed='rgbcmykw';
    color=fixed(c-1);
end
if ~isempty(coor)
    xcoor=coor(:,1);
    ycoor=coor(:,2);

```

```

L1=max(xcoor)-min(xcoor);
if ~isempty(coor2)
    xcoor2=coor2(:,1);
    L2=max(xcoor2)-min(xcoor2);
else
    L2=0;
end
xcoor=xcoor-ceil(L2/2);
ycoor=ycoor-ceil(L1/2);
patch(xcoor,ycoor,color);
set(handles.uitable1,'data',[xcoor ycoor]);
end

% -----
% -----
function uipushtool1_ClickedCallback(hObject, eventdata, handles)
% hObject    handle to uipushtool1 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)
clc;
movegui(gcf,'center');
set(handles.edit1,'string','');
set(handles.popupmenu1,'value',1);
cla(handles.axes1,'reset');
axes(handles.axes1);
axis off
set(handles.uitable1,'data',[]);

% -----
% -----
function uipushtool2_ClickedCallback(hObject, eventdata, handles)
% hObject    handle to uipushtool2 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)
clc;
frame=getframe(handles.axes1);
image=frame2im(frame);
[name_file,name_path] = uiputfile( ...
    {'*.jpg','File Type JPEG (*.jpg)';
    '*.bmp','File Type BITMAP (*.bmp)';
    '*.tif','File Type TIF (*.tif)';
    '*.png','File Type PNG (*.png)';
    '*.jpg;*.bmp;*.tif;*.png','Files of type
(*.jpg,*.bmp,*.tif,*.png)'},...
    'Save As Image');
if name_file~=0
    imwrite(image,fullfile(name_path,name_file));
end

% --- Executes on selection change in popupmenu1.
function popupmenu1_Callback(hObject, eventdata, handles)
% hObject    handle to popupmenu1 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB

```

```
% handles      structure with handles and user data (see GUIDATA)

% Hints: contents = cellstr(get(hObject,'String')) returns
popupmenu1 contents as cell array
%      contents{get(hObject,'Value')} returns selected item from
popupmenu1

% --- Executes during object creation, after setting all
properties.
function popupmenu1_CreateFcn(hObject, eventdata, handles)
% hObject      handle to popupmenu1 (see GCBO)
% eventdata    reserved - to be defined in a future version of
MATLAB
% handles      empty - handles not created until after all
CreateFcns called

% Hint: popupmenu controls usually have a white background on
Windows.
%      See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
```