

Perbandingan Algoritma *True Random*, 7 Bag, dan 35 Pools With 6 Rolls Untuk Pengacakan *Tetromino* Dalam Menangani Drought dan Flood Pada Permainan Tetris

*Diajukan Sebagai Syarat Untuk Menyelesaikan
Pendidikan Program Strata-1 Pada Jurusan Teknik Informatika
Fakultas Ilmu Komputer Universitas Sriwijaya*



Oleh :

Muhammad Razin Anggiardi
NIM : 09021281419047

**JURUSAN TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER UNIVERSITAS SRIWIJAYA 2019**

LEMBAR PENGESAHAN TUGAS AKHIR


**PERBANDINGAN ALGORITMA *TRUE RANDOM*, 7 BAG, DAN
35 POOLS WITH 6 ROLLS UNTUK PENGACAKAN
TETROMINO DALAM MENANGANI DROUGHT DAN
FLOOD PADA PERMAINAN TETRIS**

Oleh :

MUHAMMAD RAZIN ANGGIARDI
NIM : 09021281419047

Palembang, Juni 2019

Mengetahui,
Pembimbing I



Rusdi Effendi, M.T.
NIP. 1671140201820005

Pembimbing II,



Kanda Januar M, S.Kom., M.IT.
NIP 1671081080901900006

Ketua Jurusan Teknik Informatika



Rifki Primartha, M.T.
NIP 197706012009121004

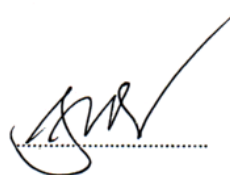
TANDA LULUS UJIAN SIDANG TUGAS AKHIR

Pada hari Senin, 24 Juni 2019 telah dilaksanakan ujian sidang tugas akhir oleh Jurusan Teknik Informatika Fakultas Ilmu Komputer Universitas Sriwijaya.

Nama : Muhammad Razin Anggiardi
NIM : 09021281419047
Judul : Perbandingan Algoritma *True Random*, *7 Bag*, dan *35 Pools With 6 Rolls* Untuk Pengacakan *Tetromino* Dalam Menangani Drought dan Flood Pada Permainan *Tetris*

1. Pembimbing I

Rusdi Efendi, M.Kom.
NIP. 1671140201820005



2. Pembimbing II

Kanda Januar M, S.Kom., M.IT.
NIP. 1671081080901900006



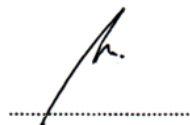
3. Penguji I

Yunita, M.Sc
NIP. 198306062015042002



4. Penguji II

Rizki Kurniati, M.T
NIP. 1671045207910003



Mengetahui,
Ketua Jurusan Teknik Informatika,



Rifkie Primartha, M.T
NIP. 197706012009121004

iii

HALAMAN PERNYATAAN

Yang bertanda tangan di bawah ini :

Nama : Muhammad Razin Anggiardi
NIM : 09021281419047
Program Studi : Teknik Informatika Bilingual
Judul Skripsi : Perbandingan Algoritma *True Random*, *7 Bag*, dan *35 Pools With 6 Rolls* Untuk Pengacakan *Tetromino* Dalam Menangani Drought dan Flood Pada Permainan *Tetris*
Hasil Pengecekan Software *iThenticate/Turnitin* : 18 %

Menyatakan bahwa Laporan Projek saya merupakan hasil karya sendiri dan bukan hasil penjiplakan/plagiat. Apabila ditemukan unsur penjiplakan/plagiat dalam laporan projek ini, maka saya bersedia menerima sanksi akademik dari Universitas Sriwijaya sesuai dengan ketentuan yang berlaku.

Demikian, pernyataan ini saya buat dengan sebenarnya dan tidak ada paksaan oleh siapapun.

Palembang, Juli 2019



(Muhammad Razin Anggiardi)

NIM. 09021281419047

MOTTO DAN PERSEMBAHAN

“Allah tidak membebani seseorang melainkan sesuai dengan kesanggupannya. Ia mendapat pahala (dari kebajikan) yang diusahakannya dan ia mendapat siksa (dari kejahatan) yang dikerjakannya.”

(Al-Baqarah: 286)

“Sesungguhnya sesudah kesulitan itu ada kemudahan.”

(Al-Insyirah: 6)

“Maka nikmat Tuhanmu yang manakah yang kamu dustakan?.”

(Ar-Rahman)

“Don’t be yourself, be your best self.”

(Kelas Cinta)

“I, Giorno Giovanna, have a dream, that I know is just!”

(Giorno Giovanna, oleh Hirohiko Araki)

Kupersembahkan karya tulis ini kepada :

- Allah SWT dan Rasulullah SAW
- Kedua Orang Tuaku
- Keluarga Besarku
- Dosen Pembimbingku dan Pengujiku
- Sahabat – sahabatku
- Almamaterku

COMPARISON OF *TRUE RANDOM*, *7 BAG*, AND *35 POOL WITH 6 ROLLS* ALGORITHM FOR THE RANDOMIZATION OF *TETROMINO* IN HANDLING DROUGHT AND FLOOD FOR TETRIS GAME

By :
Muhammad Razin Anggiardi
09021281419047

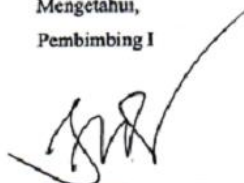
ABSTRACT

The standard or default randomization of Tetromino types, the main object in Tetris game, often feels unfair for a lot of players and causing to lose the game, such as Drought where players never get the Tetromino they need to score and continue playing, or Flood where players get one specific type of Tetromino continuously until it mess up the player's puzzle pattern. Dealing with such situation requires a special randomization with certain rules for the type of Tetromino. 7 Bag and 35 Pools With 6 Rolls are two of many algorithms that were used to randomize Tetrominos replacing True Random algorithm, which the writer tested and compared which algorithm is better at minimizing the amount of Drought and Flood between all three, the less Drought and Flood appeared then the better the algorithm is. Based by the test done, it is concluded that the 7 Bag randomizer succeeded in decreasing the amount of Drought and Flood consistently and is the superior randomizer compared to True Random and 35 Pools With 6 Rolls in Tetris Game.

Keywords: True Random, Randomization, Comparison, Game, Tetris, Tetromino, 7 Bag, 35 Pools With 6 Rolls

Palembang, Juni 2019

Mengetahui,
Pembimbing I



Rusdi Effendi, M.T.
NIP. 1671140201820005

Pembimbing II,



Kanda Januar M, S.Kom., M.IT.
NIP 1671081080901900006

Ketua Jurusan Teknik Informatika




Rifkie Primartha, M.T.
NIP 197706012009121004

**PENERAPAN ALGORITMA *ELEPHANT HERDING OPTIMIZATION*
PADA PERMASALAHAN *KNAPSACK 0-1***

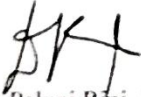
Oleh:
Hafiz Darmawan
09021281419055

ABSTRAK


Masalah *knapsack* merupakan permasalahan bagaimana memilih dari sekian banyak objek yang disimpan sehingga diperoleh suatu penyimpanan yang optimal dengan memperhatikan objek yang terdiri dari n objek (1, 2, 3,...) dimana setiap objek memiliki bobot (W_i) dan *profit* (P_i) dengan memperhatikan juga kapasitas media penyimpanan sebesar M . Terdapat banyak metode untuk menyelesaikan permasalahan *knapsack*, salah satunya menggunakan metode metaheuristik. Metode metaheuristik yang pernah digunakan untuk menyelesaikan permasalahan *knapsack 0-1* diantaranya algoritma *Binary Firefly*, algoritma *Gentika*, dan sebagainya. Terdapat algoritma metaheuristik baru yang penggunaannya belum pernah digunakan untuk masalah *knapsack* yaitu *Elephant Herding Optimization*. Penelitian ini mengimplementasikan algoritma EHO untuk menyelesaikan permasalahan *knapsack 0-1*. Setelah dilakukan pengujian pada 25 permasalahan *knapsack 0-1*, rata-rata *profit* yang dihasilkan algoritma EHO dari 6 permasalahan lebih baik dan sisanya sama daripada algoritma *Binary Firefly* yang telah dilakukan oleh penelitian sebelumnya. Nilai *profit* yang dihasilkan oleh algoritma EHO semakin baik semakin besar parameter maksimal iterasi dan populasi yang ditetapkan.

Kata kunci : *Knapsack 0-1, Elephant Herding Optimization Algorithm, Metaheuristic*


Pembimbing I


Dian Palupi Rini, M.Kom., Ph.D
NIP 197802232006042002

Palembang, Juli 2019
Pembimbing II,


Osvari Arsalan, M. T
NIP 161142806880003

Mengetahui,
Ketua Jurusan Teknik Informatika,


Rifkie Pradhartha, M. T
NIP 197706012009121004

KATA PENGANTAR



Alhamdulillah Robbil'Alamin, segala puji dan syukur kepada Allah SWT atas berkat rahmat dan karunia-Nya. Alhamdulillah Djazakumullahu Khaira, segala syukur bagi Nabi Muhammad SAW karena berkat perjuangan dan tuntunan beliau sehingga penulis dapat menyelesaikan Tugas Akhir ini dengan baik. Tugas akhir ini disusun untuk memenuhi salah satu syarat guna menyelesaikan pendidikan program Strata – 1 Program Studi Teknik Informatika pada Fakultas Ilmu Komputer di Universitas Sriwijaya.

Dalam menyelesaikan Tugas Akhir ini banyak pihak yang telah memberikan bantuan dan dukungan baik secara langsung maupun secara tidak langsung. Untuk itu Penulis ingin menyampaikan rasa terima kasih yang tulus kepada:

1. Ayah dan Ibu tercinta, H. Muhammad Ridwan dan Hj. Rita Kirana, kedua kakakku, Drg. Aditya Ramdani dan Astriana Rizki Putri S.E., M.M., serta seluruh keluarga besarku yang selalu senantiasa mendoakan, menasihati, memberikan motivasi dan dukungan luar biasa baik moril maupun materil kepada penulis;
2. Bapak Jaidan Jauhari, S.Pd., M.T. selaku Dekan Fakultas Ilmu Komputer;
3. Bapak Rifkie Primartha, M.T. selaku Ketua Jurusan Teknik Informatika;
4. Bapak Rusdi Efendi, M.Kom. selaku dosen pembimbing I dan Bapak Kanda Januar Miraswan, M.T. selaku dosen pembimbing II yang telah memberikan arahan, bimbingan, ilmu pengetahuan, nasihat serta mempermudah penulis dalam proses pengerjaan;
5. Ibu Yunita, M.Sc. selaku dosen penguji I dan Ibu Rizki Kurniati, M.T. selaku dosen penguji II yang telah memberikan masukan dan ilmu pengetahuan kepada penulis;
6. Ibu Rizki Kurniati, M.T. selaku dosen pembimbing akademik;
7. Ibu Anggina Primanita, M.IT selaku dosen pengajar mata kuliah Pemrograman Game yang menjadi inspirasi pembuatan judul tugas akhir ini;
8. Seluruh Bapak dan Ibu Dosen Program Studi Teknik Informatika dan staf Fakultas Ilmu Komputer Universitas Sriwijaya yang telah membantu dalam kelancaran penulis selama masa kegiatan perkuliahan;
9. Sahabat-sahabat penulis selama perkuliahan seluruh anggota IF Bilingual 2014 yang telah banyak membantu, mengukir cerita dan bekerja sama dengan penulis;

10. Sahabat – sahabat penulis diluar perkuliahan Dana Rizky Anastasia, Muhammad Haza Delano, Hafiz Darmawan, serta seluruh anggota Palembang Duelist Community dan Game Fortress sebagai zona ternyaman dan sumber kebahagiaan penulis dalam berbagi canda tawa serta yang selalu mampu diandalkan;

11. Semua pihak yang tidak dapat penulis sebutkan satu-persatu yang telah banyak membantu dan berperan bagi penulis terutama dalam penyelesaian tugas akhir ini, terima kasih banyak atas semuanya.

Penulis menyadari dalam penyusunan Tugas Akhir ini masih terdapat banyak kekurangan disebabkan keterbatasan pengetahuan dan pengalaman. Oleh karena itu kritik dan saran yang membangun sangat diharapkan, semoga Allah SWT selalu melimpahkan Rahmat dan Hidayah-Nya. Akhir kata dengan segala kerendahan hati, semoga Tugas Akhir ini dapat berguna dan bermanfaat bagi kita semua.

Palembang, Juni 2019

Muhammad Razin Anggiardi

DAFTAR ISI

Halaman

HALAMAN JUDUL	i
LEMBAR PENGESAHAN TUGAS AKHIR	ii
HALAMAN TANDA LULUS UJIAN SIDANG TUGAS AKHIR	iii
HALAMAN PERNYATAAN	iv
MOTO DAN PERSEMBAHAN	v
ABSTRACT	vi
ABSTRAK	vii
KATA PENGANTAR	viii
DAFTAR ISI	x
DAFTAR GAMBAR	xiv
DAFTAR TABEL	xvii

BAB 1 PENDAHULUAN

1.1 Pendahuluan	I-1
1.2 Latar Belakang	I-1
1.3 Rumusan Masalah	I-4
1.4 Tujuan Penelitian	I-4
1.5 Manfaat Penelitian	I-4
1.6 Batasan Masalah	I-5
1.7 Sistematika Penulisan	I-5
1.8 Kesimpulan	I-6

BAB II KAJIAN LITERATUR

2.1	Pendahuluan	II-1
2.2	Landasan Teori	II-1
2.3	<i>Game</i>	II-3
2.3.1	Elemen Dasar <i>Game</i>	II-4
2.3.2	MDA (<i>Mechanic/Dynamics/Aesthetics</i>)	II-5
2.3.3	<i>Game Engine</i>	II-6
2.3.4	<i>Tetris</i>	II-6
	2.4.4.1 <i>Tetromino</i>	II-8
	2.4.4.2 <i>Drought dan Flood</i>	II-9
2.3.5	<i>Play Testing</i>	II-13
2.4	Algoritma <i>True Random</i>	II-13
2.5	Algoritma <i>7 Bag</i>	II-15
2.6	Algoritma <i>35 Pools With 6 Rolls</i>	II-17
2.7	<i>Rational Unified Process</i> (RUP)	II-19
2.8	Penelitian Terkait	II-22
2.9	Kesimpulan	II-25

BAB III METODOLOGI PENELITIAN

3.1	Pendahuluan	III-1
3.2	Unit Penelitian	III-1
3.3	Pengumpulan Data	III-1
3.3.1	Jenis Data	III-1
3.3.2	Sumber Data	III-1
3.4	Tahapan Penelitian	III-2
3.4.1	Diagram Blok Proses Umum Perangkat Lunak	III-2
3.4.2	<i>Flowchart</i> Implementasi <i>True Random</i>	III-4
3.4.3	<i>Flowchart</i> Implementasi <i>7 Bag</i>	III-5
3.4.4	<i>Flowchart</i> Implementasi <i>35 Pools With 6 Rolls</i>	III-6
3.4.5	Format Data Pengujian	III-7
3.5	Metode Pengembangan Perangkat Lunak	III-8
3.6	Penjadwalan Penelitian	III-11

BAB IV PENGEMBANGAN PERANGKAT LUNAK

4.1	Pendahuluan	IV-1
4.2	Fase Insepsi	IV-1
4.2.1	Pemodelan Bisnis	IV-1
4.2.2	Kebutuhan Sistem	IV-2
a.	Fitur <i>Game Play</i>	IV-3
b.	Fitur Statistik	IV-3
4.2.3	Analisis dan Desain	IV-3
4.2.3.1	Analisis Perangkat Lunak	IV-3
a.	Analisis Kebutuhan Perangkat Lunak	IV-4
b.	Analisis Data	IV-4
4.2.3.2	Desain Perangkat Lunak	IV-5
1.	Model <i>Use Case</i>	IV-5
2.	Diagram Aktivitas	IV-9
4.3	Fase Elaborasi	IV-10
4.3.1	Pemodelan Bisnis	IV-11
4.3.1.1	Perancangan Data	IV-11
4.3.1.2	Perancangan Antarmuka	IV-12
4.3.2	Kebutuhan Sistem	IV-15
4.3.3	Diagram <i>Sequence</i>	IV-16
4.4	Fase Konstruksi	IV-19
4.4.1	Diagram Kelas	IV-19
4.4.2	Implementasi	IV-21
4.4.2.1	Implementasi Kelas	IV-21
4.4.2.2	Implementasi Antarmuka	IV-23
4.4.2.3	Implementasi Algoritma <i>Fisher Yates Shuffle</i>	IV-25
4.5	Fase Transisi	IV-27
4.5.1	Kebutuhan Sistem	IV-28
4.5.2	Rencana Pengujian	IV-28
4.5.2.1	Rencana Pengujian <i>Use Case Pilih Level</i>	IV-29
4.5.2.2	Rencana Pengujian <i>Use Case Melakukan Permainan</i>	IV-29
4.6	Kesimpulan	IV-30

BAB V HASIL DAN ANALISIS PENELITIAN

5.1	Pendahuluan	V-1
5.2	Hasil Penelitian	V-1
5.2.1	Konfigurasi Penelitian	V-1

5.2.2 Hasil Penelitian Algoritma <i>Fisher Yates Shuffle</i>	V-3
5.2.2.1 Hasil Penelitian <i>Level Hard</i>	V-3
5.3 Analisa Hasil Penelitian	V-6
5.4 Kesimpulan	V-14

BAB VI KESIMPULAN DAN SARAN

6.1 Pendahuluan	VI-1
6.2 Kesimpulan	VI-1
6.3 Saran	VI-2

DAFTAR PUSTAKA	xvi
-----------------------------	------------

LAMPIRAN	L1-1
-----------------------	-------------

DAFTAR GAMBAR

	Halaman
Gambar II-1. <i>Tetromino</i>	II-2
Gambar II-2 MDA dari Perspektif <i>Game Designer</i> dan Pemain (Migano, 2016)	II-5
Gambar II-3 Jenis-jenis <i>tetromino</i> dan rotasinya	II-9
Gambar II-4 Situasi <i>Drought tetromino I</i>	II-15
Gambar II-5. Situasi yang diakibatkan <i>Flood tetromino S dan Z</i> saja	II-16
Gambar III-1. Diagram Blok pada Tahapan Penelitian	III-3
Gambar III-2. <i>Flowchart</i> Implementasi <i>True Random</i>	III-4
Gambar III-3. <i>Flowchart</i> Implementasi <i>7 Bag</i>	III-5
Gambar III-4. <i>Flowchart</i> Implementasi <i>35 Pools With 6 Rolls</i>	III-6
Gambar III-5. Penjadwalan untuk Tahap Menentukan Ruang Lingkup dan Unit Penelitian.....	III-17
Gambar III-6. Penjadwalan untuk Tahap Menentukan Dasar Teori yang Berkaitan dengan Penelitian dan Menentukan Kriteria Pengujian	III-17
Gambar III-7. Penjadwalan untuk Tahap Menentukan Alat yang Digunakan untuk Pelaksanaan Penelitian Fase Insepsi	III-18
Gambar III-8. Penjadwalan untuk Tahap Menentukan Alat yang Digunakan untuk Pelaksanaan Penelitian Fase Elaborasi	III-18
Gambar III-9. Penjadwalan untuk Tahap Menentukan Alat yang Digunakan untuk Pelaksanaan Penelitian Fase Konstruksi	III-19
Gambar III-10. Penjadwalan untuk Tahap Menentukan Alat yang Digunakan untuk Pelaksanaan Penelitian Fase Transisi.....	III-19

Gambar III-11. Penjadwalan untuk Tahap Melakukan Pengujian Penelitian, Analisa Hasil Pengujian Penelitian dan Membuat Kesimpulan.....	III-20
Gambar IV-1. Diagram <i>Use Case</i>	IV-5
Gambar IV-2. Diagram Aktivitas <i>Use Case</i> Pilih <i>Level</i>	IV-10
Gambar IV-3. Diagram Aktivitas <i>Use Case</i> Melakukan Permainan	IV-11
Gambar IV-4. Rancang Antarmuka Menu Utama	IV-13
Gambar IV-5. Rancang Antarmuka Menu <i>Level</i>	IV-14
Gambar IV-6. Rancang Antarmuka <i>Stage</i> /Halaman Utama Permainan	IV-15
Gambar IV-7. Diagram <i>Sequence</i> Pilih <i>Level</i>	IV-17
Gambar IV-8. Diagram <i>Sequence</i> Melakukan Permainan	IV-18
Gambar IV-9. Diagram Kelas Perangkat Lunak	IV-20
Gambar IV-10. Antarmuka Halaman Menu Utama	IV-23
Gambar IV-11. Antarmuka Halaman Menu <i>Level</i>	IV-24
Gambar IV-12. Antarmuka <i>Stage</i> atau Halaman Utama Permainan	IV-24
Gambar IV-13. Antarmuka <i>Result</i> Permainan	IV-25
Gambar V-1. Jenis <i>Tetromino</i> dalam game.....	V-2
Gambar V-2. Kondisi pengacakan <i>7 Bag</i> dengan <i>Tetromino</i> yang selisih kemunculannya tidak pernah lebih dari satu.....	V-6
Gambar V-3. <i>Screenshot</i> Permainan Algoritma <i>True Random</i> setelah pengujian pertama untuk 100 kali pengacakan.....	V-9
Gambar V-4. Hasil Akhir Permainan Algoritma <i>True Random</i> setelah pengujian kedua untuk 100 kali pengacakan	V-10
Gambar V-5. <i>Screenshot</i> Permainan Algoritma <i>True Random</i> setelah pengujian ketiga untuk 100 kali pengacakan.....	V-10

Gambar V-6. *Screenshot* Permainan Algoritma 7 Bag setelah pengujian pertama untuk 100 kali pengacakan..... V-10

Gambar V-7. *Screenshot* Permainan Algoritma 7 Bag setelah pengujian kedua untuk 100 kali pengacakan..... V-11

Gambar V-8. *Screenshot* Permainan Algoritma 7 Bag setelah pengujian ketiga untuk 100 kali pengacakan..... V-11

Gambar V-9. *Screenshot* Permainan Algoritma 35 Pools With 6 Rolls setelah pengujian pertama untuk 100 kali pengacakan..... V-12

Gambar V-10. *Screenshot* Permainan Algoritma 35 Pools With 6 Rolls setelah pengujian kedua untuk 100 kali pengacakan V-12

Gambar V-11. *Screenshot* Permainan Algoritma 35 Pools With 6 Rolls setelah pengujian ketiga untuk 100 kali pengacakan V-13

DAFTAR TABEL

	Halaman
Tabel II-1. Contoh dari Algoritma <i>True Random</i>	II-14
Tabel II-2. Contoh dari Algoritma <i>7 Bag</i>	II-16
Tabel II-3. Contoh dari Algoritma <i>35 Pools With 6 Rolls</i>	II-19
Tabel III-1. Rancangan Tabel Hasil Perhitungan.....	III-7
Tabel III-2. Tahapan Pengembangan Perangkat Lunak Menggunakan Proses RUP III-9	
Tabel III-3. Penjadwalan Penelitian dalam Bentuk <i>Work Breakdown Structure (WBS)</i>	III-12
Tabel IV-1. Definisi Aktor <i>Use Case</i>	IV-6
Tabel IV-2. Definisi <i>Use Case</i>	IV-6
Tabel IV-3. Skenario Melakukan Permainan.....	IV-7
Tabel IV-4. Rancang Antarmuka <i>Test</i> /Halaman Utama Permainan	IV-12
Tabel IV-5. Implementasi Kelas	IV-12
Tabel IV-6. Implementasi Algoritma <i>Ture Random</i>	IV-20
Tabel IV-7. Implementasi Algoritma <i>7 Bag</i>	IV-20
Tabel IV-8. Implementasi Algoritma <i>35 Pools With 6 Rolls</i>	IV-21
Tabel IV-9. Rencana Pengujian Pilih Mode	IV-23
Tabel IV-10. Rencana Pengujian Melakukan Permainan	IV-23
Tabel V-1. Hasil <i>Drought</i> dan <i>Flood</i> Ketiga Algoritma	V-3

BAB I

PENDAHULUAN

1.1 Pendahuluan

Pada bab ini akan menjelaskan latar belakang penelitian untuk perbandingan algoritma-algoritma pengacak tipe bangun geometri yang terdiri dari 4 persegi (*tetromino*) pada permainan Tetris, selanjutnya dirumuskan ke dalam rumusan masalah, serta tujuan, manfaat, batasan masalah dan sistematika penulisan dari Bab I, Bab II, dan Bab III.

1.2 Latar Belakang Masalah

Perkembangan *game* di dunia semakin mengalami kemajuan sehingga *game* yang disajikan dalam permainan lebih kreatif dan mengasyikkan. Semakin berkembangnya dunia pembuatan *game*, maka *game* yang telah lama akan mengalami penurunan peminat. Hal ini disebabkan oleh permainan yang lama kalah menariknya dengan permainan yang baru. *Game* sendiri merupakan media hiburan yang setiap orang biasa menggunakannya sebagai salah satu cara mengisi waktu luang.

Tetris adalah *tile-matching puzzle video game* yang didesain oleh Alexey Pajitnov di tahun 1984, dan merupakan permainan dengan penjualan terbanyak sepanjang masa, hingga akhirnya disusul oleh *Minecraft* pada bulan Mei 2019 dan sekarang berada di posisi kedua sebagai permainan dengan penjualan terbanyak. Dalam permainan Tetris, pemain akan diarahkan bermain dalam sebuah ruangan persegi panjang berukuran 10 x 20 *tiles* dimana balok *tetromino* berjatuh secara bergilir. Pemain berusaha mendapatkan skor setinggi mungkin dengan menghabiskan 1 baris *tetromino* yang tersusun secara

horizontal dan game akan berakhir apabila ada *tetromino* yang melewati batas atas ruangan permainan.

Salah satu hal yang menjadi pusat perhatian dalam *game* Tetris yaitu jenis *tetromino* yang selalu acak saat muncul. Dalam permainan Tetris sendiri terdapat suatu metode pengacakan yang diterapkan ke dalam jenis *tetromino* yang dimunculkan. Menurut Irawan (2016), fungsi *random* yang diterapkan pada *game* sering menghasilkan beberapa data yang sama dalam pengujiannya. Berdasarkan pengujian tersebut dapat disimpulkan bahwa fungsi *random* tidak efisien dalam melakukan pengacakan, karena terdapat kemungkinan kemunculan data yang sama lebih dari satu kali dalam satu periode pengacakan atau ada satu data yang tidak pernah muncul sama sekali. Dalam permainan Tetris, selain kecerobohan pemain dalam menyusun *tetromino*, terjadinya dua skenario di atas yaitu munculnya *tetromino* yang sama (disebut *flood*) atau tidak pernahnya muncul satu *tetromino* spesifik yang dibutuhkan (disebut *drought*) dalam satu periode waktu dapat mengakibatkan kekalahan dalam permainan (Burgiel, 2016). Hal tersebut tentunya tidak disengaja oleh desainer game ini. Selain itu, fenomena *drought* dan *flood* akan lebih berimbas pada panggung olahraga *videogame* atau *e-sport* dimana suatu pemain mendapatkan *tetromino* yang lebih menguntungkan dibanding lawannya sehingga pertandingan Tetris ini menjadi lebih berorientasi kepada keberuntungan dibandingkan kemampuan dan kecekatan pemain dalam menyusun *tetromino*.

Terdapat banyak penerapan algoritma khusus dalam *game* dan fungsi *random* yang digunakan sebagai pengacakan dalam sebuah permainan baik itu untuk pengocokan kartu, pengacakan soal, posisi NPC (*Non-Playable Character*) hingga penerapan dalam pengacakan *random* pecahan puzzle. Algoritma pengacakan yang paling sederhana dan

pertama kali digunakan dalam Tetris yaitu *True Random*, dimana satu dari tujuh balok *tetromino* langsung diberi secara acak tanpa kondisi apapun. Menurut Simon Larocche (2018), pengacakan *tetromino* yang mentah seperti inilah yang mengakibatkan terjadinya *drought* dan *flood* yang parah sehingga, walau sangat langka, memungkinkan terjadinya alur permainan yang tidak dapat dimainkan. Selain itu, ada juga algoritma *7 Bag* dengan konsep “kantong” berbentuk array diisi oleh tujuh jenis *tetromino* yang diacak dalam interval tertentu secara berkala dan *35 Pools With 6 Rolls* dengan konsep “kolam” array yang diisi oleh *tetromino* dengan jumlah kelipatan angka 7 (dalam kasus ini 35) dan setiap ada *tetromino* yang terpakai, *tetromino* tersebut diubah bentuknya menjadi *tetromino* yang paling jarang muncul dalam periode permainan tersebut, dimana kedua algoritma tersebut merupakan algoritma terbaru yang digunakan dalam permainan Tetris.

Berdasarkan penelitian yang telah dilakukan sebelumnya, maka dari itu konsep permainan ini dirancang menggunakan algoritma *True Random*, *7 Bag*, dan *35 Pools With 6 Rolls* pada permainan yang mirip dengan Tetris, lalu ketiga algoritma tersebut akan diterapkan dalam *Game* serta diuji dan dibandingkan satu sama lainnya dalam menangani masalah *drought* dan *flood* dalam Tetris.

1.3 Perumusan Masalah

Berdasarkan latar belakang masalah di atas, permasalahan yang akan dibahas dalam penelitian ini adalah sebagai berikut:

1. Bagaimana mengembangkan sebuah *game* yang menyerupai Tetris dan mengimplementasikan algoritma *True Random*, *7 Bag*, dan *35 Pools With 6 Rolls* sebagai algoritma pengacak *tetromino*?

2. Seberapa baik ketiga algoritma tersebut dalam menangani masalah *drought* dan *flood* dalam permainan Tetris?

1.4 Tujuan Penelitian

Adapun tujuan penelitian ini adalah sebagai berikut:

1. Mengembangkan sebuah *game* yang menyerupai Tetris dan mengimplementasikan algoritma *True Random*, *7 Bag*, dan *35 Pools With 6 Rolls* sebagai algoritma pengacak *tetromino*.
2. Meneliti dan membandingkan ketiga algoritma tersebut dalam meminimalisir *drought* dan *flood* pada permainan Tetris.

1.5 Manfaat Penelitian

Adapun manfaat dari penelitian ini adalah sebagai berikut:

1. Bagi Dunia Pendidikan

Penelitian ini diharapkan dapat menambah informasi bagi peneliti selanjutnya dalam karya ilmiah yang berkaitan dengan implementasi algoritma kedalam permainan Tetris, dan memahami algoritma *True Random*, *7 Bag*, dan *35 Pools With 6 Rolls* secara lebih mendalam.

2. Bagi *Game Developer*

Penelitian ini diharapkan dapat digunakan pengembang game lainnya sebagai referensi pengembangan permainan berbasis unity 2D.

3. Bagi Dunia *e-Sports*

Penyelenggara e-sports dapat menggunakan penelitian ini sebagai pertimbangan dalam menciptakan aturan metagame yang baru.

1.6 Batasan Masalah

Batasan masalah dalam penelitian ini sebagai berikut :

1. *Game* akan menggunakan ruang bermain atau *playfield* berukuran 10 x 20 *tiles*, dimana *playfield* berukuran 10 x 20 tersebut dibatasi 52 buah *grid*.
2. *Game* tidak memiliki fitur skor, level dan tingkat kesulitan.
3. Penelitian dilaksanakan dengan melakukan 3000 kali pengacakan dengan interval sebagai berikut: 100 x 3, 200 x 3, 300 x 3, dan 400 x 3.

1.7 Sistematika Penulisan

Sistematika penulisan tugas akhir ini adalah sebagai berikut:

BAB I. PENDAHULUAN

Pada Bab I menjelaskan tentang latar belakang masalah, perumusan masalah, tujuan dan manfaat penelitian, batasan masalah, serta sistematika penulisan.

BAB II. KAJIAN LITERATUR

Pada Bab II menjelaskan dasar-dasar teori yang akan digunakan dalam penelitian, seperti pengetahuan dasar tentang permainan dan algoritma yang akan digunakan dalam penelitian.

BAB III. METODOLOGI PENELITIAN

Pada Bab III menjelaskan tentang unit penelitian, metode pengumpulan data pada penelitian yang terbagi menjadi jenis data, sumber data, dan teknik pengumpulan data, selanjutnya penjelasan langkah-langkah pengumpulan data, tahapan pada penelitian, metode pengembangan perangkat lunak, dan manajemen proyek penelitian dalam bentuk tabel penjadwalan.

1.8 Kesimpulan

Pada bab ini telah dibahas mengenai penelitian yang akan dilaksanakan yaitu penelitian mengenai perbandingan algoritma *True Random*, *7 Bag*, dan *35 Pools With 6 Rolls* yang akan diimplementasikan pada *tetromino* sebagai pengacak jenisnya dalam permainan Tetris yang bertujuan untuk meneliti algoritma mana yang paling efektif dalam menangani *flood* dan *drought* pada permainan Tetris. Selanjutnya, teori-teori yang berkaitan dengan penelitian akan dibahas pada Bab II.

DAFTAR PUSTAKA

- Aleem, S., L.F. Capretz. dan F. Ahmed. 2016. *Game Development Software Engineering Process Life Cycle: A Simantic Review*. *Journal of Software Engineering Research and Development* (2016) 4:6.
- Burgiel, Heidi. 1996. *How to lose at Tetris*. Department of Mathematics, Statistics, and Computer Science, m/c 249, University of Illinois at Chicago, 851 South Morgan St., USA.
- Classic Tetris Twitch Channel (<https://www.twitch.tv/classictetris/>, diakses 7 Juni 2019)
- Classic Tetris World Championship Website (<https://thectwc.com/>, diakses 7 Juni 2019)
- Classic Tetris Youtube Channel (<https://www.youtube.com/channel/UC-8BAEcWSEs4-KQp1ULVnaQ>, diakses 15 Februari 2019)
- Cornelius, Adam. (Producer) 2011. *Ecstasy of Order: The Tetris Masters*. Reclusion Films. Los Angeles, California, United States of America. 93 mins. (<http://watch.ecstasyoforder.com/>, diakses 8 Juni 2019)
- Farisi, A. 2015. Analisis Perbandingan Algoritma *Fisher Yates Shuffle* dan *Naive Shuffle*. (<https://id.scribd.com/doc/269434539/Analisis-Perbandingan-Algoritma-Fisher-Yates-Shuffle-Dan-Naive-Shuffle>, diakses 15 Februari 2019).
- Fikria, Balqis Kamalia. 2014. *Implementasi Algoritma Fisher-Yates Shuffle Sebagai Pengacak Posisi Non-Playable Character (NPC) Pada Game Tantra Bahari*. Jurusan Teknik Informatika, Sains dan Teknologi. UIN Maulana Malik Ibrahim, Malang.
- Golomb, Solomon W. 1994. *Polyminoes (2nd ed.)*. Princeton, New Jersey: Princeton University Press. (ISBN 0-691-02444-8)
- Karp, R.M. 1991. *An Introduction to Randomized Algorithms*. *Discrete Applied Mathematics* 34(1991):165 – 201.
- Kruchten, P. 2005. *The Rational Unified Process: An Introduction (2nd Edition)*. Addison – Wesley Longman Publishing Co., Inc, Boston, MA, USA.
- Laroche, Simon. 2018. *The history of Tetris randomizers*. (<https://simon.lc/the-history-of-tetris-randomizers> , diakses 4 Februari 2019).

- Levy, L. dan J. Novak. 2010. *Game Development Essentials: Game QA & Testing*. Delmar, Cengage Learning. Clifton Park, USA.
- Migano, Marco. 2016. *Mechanics Dynamics Aesthetics (MDA): The Game Design Theory Behind Games*. (<http://gamedevopertips.com/mechanics-dynamics-aesthetics-game-design-theory-behind-games/> , diakses 2 Februari 2019).
- NES Tetris 81 Drought (<https://www.youtube.com/watch?v=1hIlqkwUcy0> , diakses 25 Februari 2019).
- Rouse, R. 2004. *Game Design: Theory and Practice* (2nd Edition). Wordware Publishing, Inc, Los Rios Boulevard Plano, Texas, USA.
- Sagala, M.L., E.M.A. Jonemaro. dan W.S Wardhono. 2017. Pengembangan *Game Platformer 2D* Menggunakan Teknik *Projection Mapping*. Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer 1(11):1160 – 1168.
- Singh, P., A. Batheja. dan A. Chowdhury. 2015. *Predictive Music Shuffling Algorithm*. *International Journal of Computer Science and Information Technologi* 6(5): 4159 – 4162.
- Spiel, K. *et al.* 2017. “Not another Z piece!” *Adaptive Difficulty in TETRIS*. CHI 2017, May 6-11, 2017, Denver, CO, USA.
- Wibowo, I. 2014. Pembuatan *Game* Edukasi “Negara” untuk Memperkenalkan Negara di Dunia Kepada Anak Usia Dini Berbasis Android. Naskah Publikasi Sekolah Tinggi Manajemen Informatika dan Komputer AMIKOM Yogyakarta.