

HALAMAN JUDUL

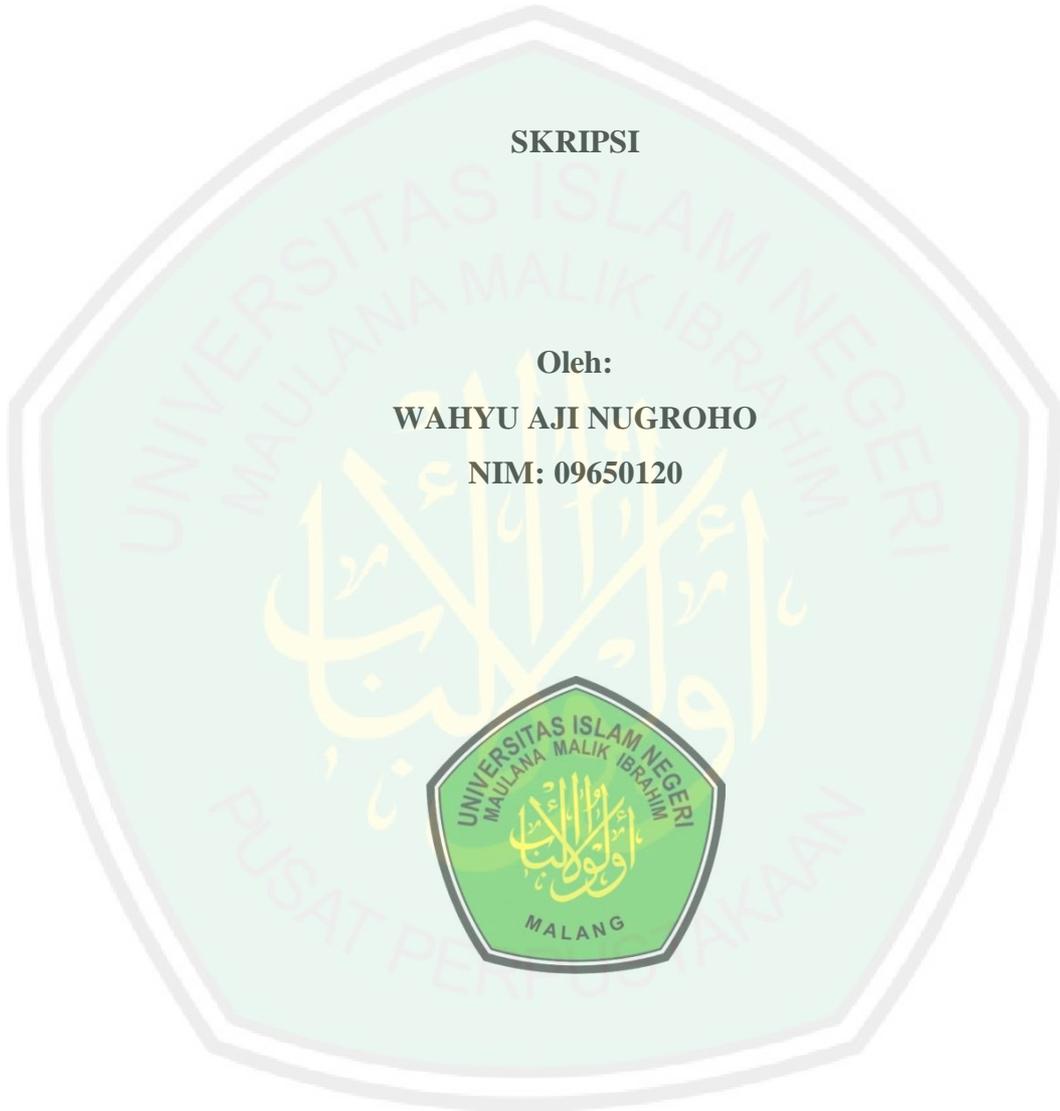
**APLIKASI PERMAINAN PAIRS BAHASA ARAB MENGGUNAKAN
ALGORITMA DEPTH FIRST SEARCH DAN MULTIPLICATIVE CRNG**

SKRIPSI

Oleh:

WAHYU AJI NUGROHO

NIM: 09650120



**JURUSAN TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI (UIN) MAULANA MALIK IBRAHIM
MALANG
2013**

HALAMAN PENGAJUAN

**APLIKASI PERMAINAN PAIRS BAHASA ARAB MENGGUNAKAN
ALGORITMA DEPTH FIRST SEARCH DAN MULTIPLICATIVE CRNG**

SKRIPSI

**Diajukan kepada:
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang
Untuk Memenuhi Salah Satu Persyaratan Dalam
Memperoleh Gelar Sarjana Komputer (S.Kom)**

Oleh:

**WAHYU AJI NUGROHO
NIM: 09650120**

**JURUSAN TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI (UIN) MAULANA MALIK IBRAHIM
MALANG
2013**

HALAMAN PERSETUJUAN

**APLIKASI PERMAINAN PAIRS BAHASA ARAB MENGGUNAKAN
ALGORITMA DEPTH FIRST SEARCH DAN MULTIPLICATIVE CRNG**

SKRIPSI

Oleh :

Nama : Wahyu Aji Nugroho
NIM : 09650120
Jurusan : Teknik Informatika
Fakultas : Sains Dan Teknologi

Telah Disetujui, 4 April 2013

Dosen Pembimbing I

Dosen Pembimbing II

Hani Nurhavati, M.T
NIP. 19780625 200801 2 006

Fachrul Kurniawan, M.MT
NIP. 19771020 200901 1 001

Mengetahui,
Ketua Jurusan Teknik Informatika

Ririen Kusumawati, S.Si, M.Kom
NIP. 19720309 200501 2 002

HALAMAN PENGESAHAN

**APLIKASI PERMAINAN PAIRS BAHASA ARAB MENGGUNAKAN
ALGORITMA DEPTH FIRST SEARCH DAN MULTIPLICATIVE CRNG**

SKRIPSI

Oleh :

**Wahyu Aji Nugroho
NIM. 09650120**

Telah Dipertahankan Di Depan Dewan Penguji Skripsi
Dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan
Untuk Memperoleh Gelar Sarjana Komputer (S.Kom)

Tanggal 13 April 2013

Susunan Dewan Penguji:	Tanda Tangan
1. Penguji Utama : <u>Fressy Nugroho, M.T</u> NIP. 19710722 201101 1 001	()
2. Ketua Penguji : <u>A'la Syauqi, M.Kom</u> NIP. 19771201 200801 1 007	()
3. Sekretaris : <u>Hani Nurhayati, M.T</u> NIP. 19780625 200801 2 006	()
4. Anggota Penguji : <u>Fachrul Kurniawan, M.MT</u> NIP. 19771020 200901 1 001	()

Mengetahui,
Ketua Jurusan Teknik Informatika

**Ririen Kusumawati, S.Si, M.Kom
NIP. 19720309 200501 2 002**

HALAMAN PERNYATAAN
ORISINALITAS PENELITIAN

Saya yang bertandatangan di bawah ini:

Nama : Wahyu Aji Nugroho
NIM : 09650120
Fakultas/Jurusan : Sains Dan Teknologi / Teknik Informatika
Judul Penelitian : Aplikasi Permainan Pairs Bahasa Arab
Menggunakan Algoritma Depth First Search Dan
Multiplicative CRNG

Menyatakan dengan sebenarnya bahwa skripsi yang saya tulis ini benar-benar merupakan hasil karya saya sendiri, bukan merupakan pengambil alihan data, tulisan atau pikiran orang lain yang saya akui sebagai hasil tulisan atau pikiran saya sendiri, kecuali dengan mencantumkan sumber cuplikan pada daftar pustaka. Apabila di kemudian hari terbukti atau dapat dibuktikan skripsi ini hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut

Malang, 4 April 2013
Yang Membuat Pernyataan,

Wahyu Aji Nugroho
09650120

HALAMAN MOTTO

JIKA ADA KEMAUAN PASTI ADA JALAN

**DI SETIAP KESULITAN
PASTI ADA KEMUDAHAN**

فَإِنَّ مَعَ الْعُسْرِ يُسْرًا

"Karena sesungguhnya sesudah kesulitan itu ada kemudahan."

إِنَّ مَعَ الْعُسْرِ يُسْرًا

"Sesungguhnya sesudah kesulitan itu ada kemudahan."

(QS. Alam Nasyroh:5- 6)

HALAMAN PERSEMBAHAN

Pada halaman ini saya akan mengucapkan banyak terima kasih kepada semua pihak yang telah mendukung saya sehingga penelitian ini selesai:

1. *Allah SWT*
2. *Muhammad SAW*
3. *Bapak/ibuk ku, Suyanto dan Loeki Koeswardani*
4. *Mbak ku Nurvia Ariyanti*
5. *Keluarga besar mbah Toesiran*
6. *Temen-temen kuliah*
7. *Temen-temen kontrakan :*
(febri,oki,bahtiar,lutfi,tino,geby,habib,salis,syauqil)
8. *Temen-temen kosmea :*
(sam wildan,sam niam,sam novan,sam budi,bayu,mbk bela,mbk shinta,mbk suci)
9. *Temen-temen FUTSAL TI 2009*
10. *Idhar/mantan arek kontrakan <yang bantuin desain>*
11. *Dan semua saja yang gak bisa disebutin, suwoon pool ker!*

KATA PENGANTAR



Segala puji bagi Allah SWT yang telah melimpahkan rahmat serta karuniaNya kepada penulis sehingga bisa menyelesaikan skripsi dengan judul “Aplikasi Permainan Pairs Bahasa Arab Menggunakan Algoritma Depth First Search Dan Multiplicative CRNG” dengan baik.

Shalawat serta salam semoga tercurah kepada Nabi Agung Muhammad SAW yang telah membimbing umatnya dari gelapnya kekufuran menuju cahaya Islam yang terang benderang.

Penulis menyadari keterbatasan pengetahuan yang penulis miliki, karena itu tanpa keterlibatan dan sumbangsih dari berbagai pihak, sulit bagi penulis untuk menyelesaikan skripsi ini. Maka dari itu dengan segenap kerendahan hati patutlah penulis ucapkan terima kasih kepada:

1. Prof. DR. H. Imam Suprayogo, selaku rektor UIN Maulana Malik Ibrahim Malang, yang telah banyak memberikan pengetahuan dan pengalaman yang berharga.
2. Prof. Drs. Sutiman Bambang Sumitro, SU., D.Sc selaku Dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.
3. Hani Nurhayati, M.T dan Fachrul Kurniawan, M.MT selaku dosen pembimbing I dan II yang telah meluangkan waktu untuk membimbing, memotivasi, mengarahkan dan memberi masukan dalam pengerjaan skripsi ini.

4. Segenap sivitas akademika Jurusan Teknik Informatika, terutama seluruh dosen, terima kasih atas segenap ilmu dan bimbingannya.
5. Ayahanda dan Ibunda tercinta yang senantiasa memberikan doa dan restunya kepada penulis dalam menuntut ilmu.
6. Kakak penulis yang selalu memberikan semangat kepada penulis untuk menyelesaikan skripsi ini.
7. Semua pihak yang tidak mungkin penulis sebutkan satu-persatu, atas segala yang telah diberikan kepada penulis dan dapat menjadi pelajaran.

Sebagai penutup, penulis menyadari dalam skripsi ini masih banyak kekurangan dan jauh dari sempurna. Semoga apa yang menjadi kekurangan bisa disempurnakan oleh peneliti selanjutnya. Apa yang menjadi harapan penulis, semoga karya ini bermanfaat bagi kita semua. Amin.

Malang, April 2013

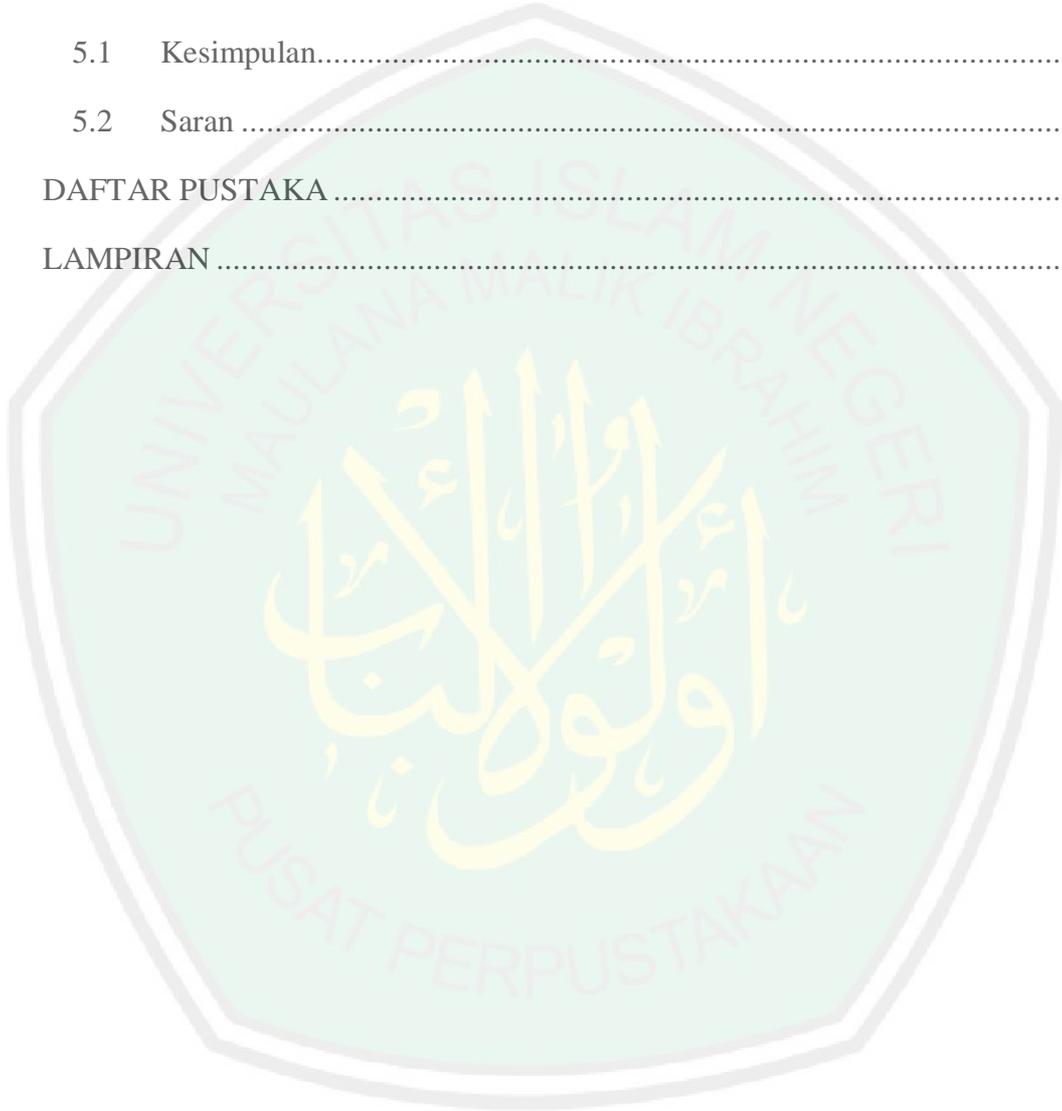
Penulis

DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PENGAJUAN	ii
HALAMAN PERSETUJUAN	iii
HALAMAN PERNYATAAN.....	v
HALAMAN MOTTO.....	vi
HALAMAN PERSEMBAHAN	vii
KATA PENGANTAR	viii
DAFTAR ISI	x
DAFTAR GAMBAR	xiii
DAFTAR TABEL	xv
ABSTRAK	xvi
ABSTRACT	xvii
BAB I PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	5
1.3 Batasan Masalah	6
1.4 Tujuan Penelitian.....	6
1.5 Manfaat Penelitian.....	6
1.6 Sistematika Penulisan	7
BAB II TINJAUAN PUSTAKA	9
2.1 Permainan.....	9
2.1.1 Sejarah Perminan.....	9
2.1.2 Klasifikasi Permaian.....	10

2.2	Bahasa Arab	12
2.3	Permainan Pairs	13
2.4	Depth First Search	14
2.5	<i>Multiplicative CRNG (Congruential Random Number Generator)</i>	17
2.6	Platform Android	20
BAB III ANALISA DAN PERANCANGAN		22
3.1	Analisa dan Perancangan Sistem.....	22
3.1.1	Keterangan Umum.....	22
3.1.2	Rancangan Kategori Permainan	23
3.1.3	Rancangan Level Permainan.....	25
3.2	Rancangan Algoritma Depth First Search dan Multiplicative CRNG....	26
3.2.1	Pencarian Solusi Permainan	26
3.2.2	Pembangkit Permainan	30
3.3	Perancangan Aplikasi.....	31
3.3.1	Antarmuka Permainan	31
3.3.2	Kebutuhan Sistem.....	39
BAB IV HASIL DAN PEMBAHASAN		40
4.1	Implementasi Algoritma Multiplicative CRNG dan Depth First Search	40
4.1.1	Multiplicative CRNG.....	40
4.1.2	Depth First Search	43
4.2	Implementasi Aplikasi	46
4.3	Uji Coba	53
4.3.1	Pembangkitan Permainan Menggunakan Algoritma Multiplicative CRNG	53
4.3.2	Pencarian Solusi Permainan Menggunakan Algoritma Depth First Search	63

4.3.3 Uji Coba Pengguna.....	72
4.4 Integrasi Game Arabic Pairs dengan Islam.....	76
BAB V PENUTUP	78
5.1 Kesimpulan.....	78
5.2 Saran	78
DAFTAR PUSTAKA	80
LAMPIRAN	82



DAFTAR GAMBAR

Gambar 2.1 Salah satu tampilan permainan Pairs	14
Gambar 2.2 Alur Algoritma Depth First Search.....	15
Gambar 3.1 Tampilan website kamus online Oxford University.....	23
Gambar 3.2 Flowchart Pencarian Solusi Permainan	27
Gambar 3.3 Bentuk matriks permainan Pairs.....	28
Gambar 3.4 Flowchart Pembangkit Permainan.....	30
Gambar 3.5 Rancangan Halaman Utama	31
Gambar 3.6 Rancangan Halaman Pilih Skor.....	32
Gambar 3.7 Rancangan Halaman Skor.....	32
Gambar 3.8 Rancangan Halaman Pilih Kategori.....	33
Gambar 3.9 Rancangan Halaman Pilih Level	34
Gambar 3.10 Rancangan Halaman Permainan.....	34
Gambar 3.11 Rancangan Halaman Entry Player Name	36
Gambar 3.12 Flowchart Permainan Arabic Pairs	37
Gambar 3.13 Class Diagram Permainan Arabic Pairs	38
Gambar 4.1 Halaman Utama	46
Gambar 4.2 Halaman Help.....	46
Gambar 4.3 Halaman Kategori.....	47
Gambar 4.4 Halaman Pilih Level	48
Gambar 4.5 Pilihan Level Permainan	48
Gambar 4.6 Halaman Permainan.....	49
Gambar 4.7 Kartu Terbuka.....	49
Gambar 4.8 Pasangan kartu ditemukan.....	51

Gambar 4.9 Form Skor.....	51
Gambar 4.10 Halaman Tabel Skor	52
Gambar 4.11 Pengujian Pembangkitan Permainan Level Easy ke 1	53
Gambar 4.12 Pengujian Pembangkitan Permainan Level Easy ke 2.....	54
Gambar 4.13 Pengujian Pembangkitan Permainan Level Easy ke 3.....	55
Gambar 4.14 Pengujian Pembangkitan Permainan Level Medium ke 1	56
Gambar 4.15 Pengujian Pembangkitan Permainan Level Medium ke 2	57
Gambar 4.16 Pengujian Pembangkitan Permainan Level Medium ke 3	58
Gambar 4.17 Pengujian Pembangkitan Permainan Level Hard ke 1.....	59
Gambar 4.18 Pengujian Pembangkitan Permainan Level Hard ke 2.....	60
Gambar 4.19 Pengujian Pembangkitan Permainan Level Hard ke 3.....	61
Gambar 4.20 Pengujian Pencarian Solusi Level Easy ke 1	63
Gambar 4.21 Pengujian Pencarian Solusi Level Easy ke 2	64
Gambar 4.22 Pengujian Pencarian Solusi Level Easy ke 3	65
Gambar 4.23 Pengujian Pencarian Solusi Level Medium ke 1.....	66
Gambar 4.24 Pengujian Pencarian Solusi Level Medium ke 2.....	67
Gambar 4.25 Pengujian Pencarian Solusi Level Medium ke 3.....	68
Gambar 4.26 Pengujian Pencarian Solusi Level Hard ke 1	69
Gambar 4.27 Pengujian Pencarian Solusi Level Hard ke 2	70
Gambar 4.28 Pengujian Pencarian Solusi Level Hard ke 3	71

DAFTAR TABEL

Tabel 2.1 Isi <i>Stack</i> dalam <i>DFS</i>	16
Tabel 3.1 Kategori Permainan	24
Tabel 3.2 Jumlah kartu yang ditampilkan setiap level.....	25
Tabel 3.3 Alur Penelusuran Pencarian Solusi	28
Tabel 4.1 Pengujian Pembangkitan Permainan Level Easy ke 1	54
Tabel 4.2 Pengujian Pembangkitan Permainan Level Easy ke 2	55
Tabel 4.3 Pengujian Pembangkitan Permainan Level Easy ke 3	56
Tabel 4.4 Pengujian Pembangkitan Permainan Level Medium ke 1	57
Tabel 4.5 Pengujian Pembangkitan Permainan Level Medium ke 2.....	58
Tabel 4.6 Pengujian Pembangkitan Permainan Level Medium ke 3.....	59
Tabel 4.7 Pengujian Pembangkitan Permainan Level Hard ke 1	60
Tabel 4.8 Pengujian Pembangkitan Permainan Level Hard ke 2	61
Tabel 4.9 Pengujian Pembangkitan Permainan Level Hard ke 3	62
Tabel 4.10 Pengujian Pencarian Solusi Level Easy ke 1	63
Tabel 4.11 Pengujian Pencarian Solusi Level Easy ke 2	64
Tabel 4.12 Pengujian Pencarian Solusi Level Easy ke 3	65
Tabel 4.13 Pengujian Pencarian Solusi Level Medium ke 1.....	66
Tabel 4.14 Pengujian Pencarian Solusi Level Medium ke 2.....	67
Tabel 4.15 Pengujian Pencarian Solusi Level Medium ke 3.....	68
Tabel 4.16 Pengujian Pencarian Solusi Level Hard ke 1	69
Tabel 4.17 Pengujian Pencarian Solusi Level Hard ke 2.....	70
Tabel 4.18 Pengujian Pencarian Solusi Level Hard ke 3.....	71
Tabel 4.19 Rekapitulasi Hasil Uji Coba.....	73

ABSTRAK

Nugroho, Wahyu Aji. 2013. **Aplikasi Permainan Pairs Bahasa Arab Menggunakan Algoritma Depth First Search Dan Multiplicative CRNG.** Skripsi. Jurusan Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.
Pembimbing: (I) Hani Nurhayati, M.T (II) Fachrul Kurniawan, M.MT

Kata Kunci: *Pairs, Bahasa Arab, Depth First Search, Multiplicative CRNG, Pembangkit, Penyelesai*

Bahasa Arab merupakan bahasa Alquran yang perlu kita pelajari. Banyak sekali media pembelajaran bahasa Arab. Salah satunya melalui permainan interaktif. Permainan Pairs merupakan salah satu permainan yang interaktif. Pairs merupakan permainan yang mempunyai aturan cukup sederhana. Pemain diminta menemukan pasangan kartu-kartu bergambar dalam keadaan kartu tertutup pada awal permainannya. Jika kartu pertama dan kedua yang dipilih mempunyai gambar yang sama maka kartu tersebut akan hilang, jika tidak maka akan tertutup kembali. Permainan akan selesai jika pemain berhasil menemukan semua pasangan kartu dari kartu yang muncul di awal permainan. Penelitian kali ini, penulis menambahkan algoritma untuk pembangkitan permainan menggunakan algoritma *Multiplicative CRNG*, untuk penyelesai permainan masih menggunakan algoritma *DFS* dan dibangun pada *platform* Android. Permainan Pairs diujikan pada perangkat *mobile* yang menggunakan *platform* Android. Permainan Pairs ini juga dibangun untuk permainan edukasi dengan menambahkan pembelajaran pengenalan dan pengucapan kosakata bahasa Arab di dalamnya. Hasil uji coba penelitian ini, dapat disimpulkan bahwa algoritma *DFS* dan *Multiplicative CRNG* dapat diimplementasikan pada permainan Pairs sebagai pencarian solusi dan pembangkit permainan, selain itu game interaktif Pairs menggunakan bahasa Arab dapat membantu siswa-siswi Sekolah Dasar maupun Madrasah Ibtidaiyah dalam belajar bahasa arab.

ABSTRACT

Nugroho, Wahyu Aji. 2013. **Application Arabic Pairs Game Using Depth First Search and Multiplicative CRNG Algorithms**. Thesis. Informatics Department of Faculty of Science and Technology. Maulana Malik Ibrahim State Islamic University, Malang.

Adviser: (I) Hani Nurhayati, M.T (II) Fachrul Kurniawan, M.MT

Keywords: *Pairs, Arabic, Depth First Search, Multiplicative CRNG, Generator, Solver*

Arabic is the Alquran language that need to be learn. There are Lot of learning Arabic media. interactive games is one of the learning Arabic media. Pairs game is one of the interactive game. Pairs game has simple rules. Players are asked to find a picture cards partner in an enclosed card at the beginning of the game. if the first and second selected card has the same image, the card will be disappeared, otherwise it will be closed again. The game will be completed if the player succeeded to find all pairs of cards from a card that appears at the beginning of the game. Study, the authors add algorithm for generating games using algorithms Multiplicative CRNG, algorithm DFS used to resolve the game and is't built on the Android platform. Pairs game tested on mobile devices that using the Android platform. Pairs game is built for educational games by adding learning vocabulary recognition and pronunciation of the Arabic language in it. The trial results of this study, it can be concluded that the algorithm DFS and Multiplicative CRNG Pairs can be implemented in the game as the solutions and generate games, moreover interactive games Pairs in Arabic can help elementary school students and Madrasah Ibtidaiah in learning Arabic.



BAB I

PENDAHULUAN

1.1 Latar Belakang

Permainan atau game adalah implementasi dari bidang ilmu komputer. Namun dalam perkembangannya game sudah banyak diimplementasikan dalam perangkat mobile. Perkembangan permainan pada masa kini sudah sangat pesat dan telah menjadi *tren* tersendiri di dunia karena sebagian besar pengguna perangkat mobile dipastikan menginstall aplikasi game di dalamnya.

Aplikasi permainan selain digunakan untuk menghilangkan kejenuhan juga digunakan sebagai media pembelajaran, salah satunya adalah pembelajaran bahasa Arab.

Bahasa menurut pengertian bahasa, adalah sebagaimana yang dikemukakan oleh Syekh Al-Ghulayaini berikut ini:

“Bahasa itu adalah lafal-lafal yang dengannya setiap bangsa mengungkapkan maksud-maksud mereka” (Jami'ud Durus Al-Arabiyah, 1980:4)

Menurut Abubakar Muhammad, bahwa bahasa itu alat untuk mengungkapkan isi hati kepada orang lain. Bahasa itu banyak sekali, sesuai dengan banyaknya suku bangsa dan bangsa itu sendiri. Hal itu menjadi bukti kekuasaan Allah SWT. (1996:3)

Sebagaimana dijelaskan dalam surat Ar-Rum (Q.S. 30), ayat 22:

وَمِنْ آيَاتِهِ خَلْقَ السَّمَوَاتِ وَالْأَرْضِ وَاخْتِلَافُ أَلْسِنَتِكُمْ وَأَلْوَانِكُمْ إِنَّ فِي
ذَلِكَ لَآيَاتٍ لِّلْعَالَمِينَ

Artinya:

Dan di antara tanda-tanda kekuasaan-Nya ialah menciptakan langit dan bumi dan berlain-lainan bahasamu dan warna kulitmu. Sesungguhnya pada yang demikian itu benar-benar terdapat tanda-tanda bagi orang-orang yang Mengetahui.(Q.S. 30:22)

Bahasa Arab merupakan salah satu bahasa internasional yang cukup unik dibandingkan dengan bahasa-bahasa internasional yang lainnya seperti bahasa Inggris. Untuk mempelajarinya saja ada beberapa disiplin ilmu yang memiliki perbedaan. Ilmu nahwu lebih menekankan pada harakat terakhir sedangkan ilmu sharaf lebih menekankan pada perubahan bentuk kata, sedangkan ilmu balaghah cenderung mempelajari sastra arab.

Pentingnya penguasaan bahasa arab salah satunya adalah mempermudah komunikasi kita kepada Allah SWT untuk memuji dan kebesarannya dan RosulNya, termasuk di dalam hal berdoa. Bukti-bukti bahwa Al-Quran menggunakan bahasa Arab tercantum dalam firman Allah dalam (Q.S. 20), ayat 113 berikut:

وَكَذَلِكَ أَنزَلْنَاهُ قُرْآنًا عَرَبِيًّا وَصَرَّفْنَا فِيهِ مِنَ الْوَعِيدِ لَعَلَّهُمْ يَتَّقُونَ أَوْ يُحَدِّثُهُمْ
ذِكْرًا

Artinya:

Dan Demikianlah kami menurunkan Al Quran dalam bahasa Arab, dan kami Telah menerangkan dengan berulang kali, di dalamnya sebahagian dari ancaman, agar mereka bertakwa atau (agar) Al Quran itu menimbulkan pengajaran bagi mereka (Q.S. 20, ayat 113)

Selain itu dalam sebuah hadits shahih riwayat Al-Bukhari juga dijelaskan pentingnya mempelajari dan mengajarkan Al-Qur'an. Berikut hadits yang dimaksud:

عَنْ عُثْمَانَ بْنِ عَفَّانَ (رض) قَالَ قَالَ رَسُولُ اللَّهِ (ص)
خَيْرُكُمْ مَنْ تَعَلَّمَ الْقُرْآنَ وَعَلَّمَهُ .

Artinya:

Dari Usman bin 'Affan ra, ia berkata, Rasulullah SAW bersabda, "Orang yang paling baik di antara kalian adalah yang mempelajari Al-Qur'an dan mengajarkannya". (Ali Mustafa Yaqub, 1990:17)

Banyak sekali berbagai jenis permainan dalam *smartphone* mulai dari yang sederhana sampai yang kompleks. Sistem operasi Android adalah salah satu *platform* yang digunakan untuk mengembangkan aplikasi-aplikasi permainan tersebut. Salah satu permainan yang dikembangkan dalam sistem operasi Android adalah permainan *Pairs*.

Permainan *Pairs* merupakan permainan yang terdiri dari beberapa kartu gambar. Kartu-kartu tersebut mempunyai gambar yang berbeda jenisnya. Tiap satu jenis gambar terdapat dua buah kartu, jadi setiap satu jenis kartu terdapat sepasang gambar. Pada permainan *Pairs*, jumlah kartu juga dipastikan berjumlah genap karena setiap jenis gambar dipastikan mempunyai pasangan.

Pada awal permainan, semua kartu disusun secara acak membentuk suatu matriks. Semua kartu tersebut disusun dalam keadaan tertutup. Pemain mulai memilih dua buah kartu yang ingin dibuka. Jika kedua kartu tersebut mempunyai gambar yang tidak sama maka kedua kartu tersebut akan kembali tertutup. Namun jika gambar pada kedua kartu tersebut sama maka kartu tersebut akan dibiarkan terbuka. Proses ini akan dilakukan oleh pemain secara terus menerus sampai pemain menemukan kartu-kartu yang sepasang. Permainan akan berakhir jika pemain sanggup menemukan semua pasangan dari setiap jenis kartu. (Lovinta, 2007:1).

Penelitian tentang permainan Pairs pernah dilakukan oleh Lovinta Happy Atrianawati (Lovinta, 2007). Pada penelitian tersebut, aplikasi permainan Pairs dibangun berbasis aplikasi *desktop* menggunakan bahasa pemrograman *pascal*. Permainan tersebut menerapkan algoritma *Depth First Search (DFS)* untuk pencarian solusi permainan dan untuk pengacakan kartu masih menggunakan *method random* bawaan bahasa pemrograman. Hasil dari penelitian tersebut telah membuktikan bahwa algoritma *DFS* cukup cepat untuk melakukan pencarian pasangan kartu gambar, akan tetapi kekurangan aplikasi permainan tersebut adalah hanya bisa dimainkan di komputer saja.

Semua jenis permainan atau game membutuhkan cara atau algoritma untuk menyelesaikannya. Termasuk permainan Pairs ini. Algoritma sangat penting dalam hal ini. Bahkan, dapat dikatakan algoritma adalah inti dari informatika. Algoritma adalah langkah-langkah logis dalam menyelesaikan suatu masalah. Seiring dengan perkembangan kehidupan yang semakin kompleks, kebutuhan akan algoritma yang handal untuk menyelesaikan berbagai persoalan komputasi sangatlah diperlukan. Sejalan dengan algoritma, dalam Surat Ar-Rum ayat 8:

أَوَلَمْ يَتَفَكَّرُوا فِي أَنفُسِهِمْ ۗ مَا خَلَقَ اللَّهُ السَّمَوَاتِ وَالْأَرْضَ وَمَا بَيْنَهُمَا إِلَّا
بِالْحَقِّ وَأَجَلٍ مُّسَمًّى ۗ وَإِنَّ كَثِيرًا مِّنَ النَّاسِ بِلِقَائِ رَبِّهِمْ لَكَافِرُونَ ﴿٨﴾

Artinya:

Dan Mengapa mereka tidak memikirkan tentang (kejadian) diri mereka? Allah tidak menjadikan langit dan bumi dan apa yang ada diantara keduanya melainkan dengan (tujuan) yang benar dan waktu yang ditentukan. dan Sesungguhnya kebanyakan di antara manusia benar-benar ingkar akan pertemuan dengan Tuhannya. (Q.S. Ar-Rum, ayat 8)

Dalam firman Allah tersebut mengisyaratkan kepada kita bahwa setiap kejadian itu pasti ada sebab akibatnya, begitu pula dalam penelitian ini, untuk mendapatkan pencarian solusi dan membangkitkan permainan dalam *game Pairs* juga membutuhkan algoritma di dalamnya.

Penelitian ini akan menggunakan perpaduan dua buah algoritma yaitu *Depth First Search* sebagai pencarian solusi dan *Multiplicative CRNG* sebagai pembangkit permainan, dengan harapan permainan *Pairs* akan lebih cepat dan ringan dari sisi proses serta pengacakan kartu lebih variatif. Selain itu aplikasi berbasis Android ini juga lebih menguntungkan karena pemain bisa memainkan permainan ini kapanpun dan di manapun pemain berada.

Penelitian ini penulis mencoba membangun sebuah aplikasi permainan yang edukatif dengan memberikan efek suara berbahasa Arab pada saat pemain membuka kartu yang tertutup sesuai gambar yang tampak pada kartu tersebut. Penulis berharap pemain bisa sekaligus belajar mengenal bahasa Arab, yang merupakan bahasa dalam kitab suci Al Quran, hal ini adalah tujuan utama penulis dalam melakukan penelitian ini.

1.2 Rumusan Masalah

Bagaimana membangun aplikasi permainan *Pairs* bahasa Arab menggunakan algoritma *Depth First Search (DFS)* sebagai pencarian solusi dan *Multiplicative CRNG* sebagai pembangkit permainan dan bagaimana mengeluarkan suara berbahasa Arab sesuai dengan gambar pada kartu yang dipilih oleh pemain?

1.3 Batasan Masalah

Batasan masalah pada penelitian ini adalah:

- a) Permainan ini dimainkan oleh satu pemain/*single player*
- b) Kartu berupa gambar-gambar kosakata
- c) Setiap kartu yang dibuka oleh pemain akan mengeluarkan suara berbahasa Arab sesuai gambar
- d) Permainan ini ada 3 level yaitu mudah, sedang dan sulit
- e) Permainan ini ada 4 kategori, yaitu *Amusing Animal*, *At Home*, *At School*, dan *Count With Me*
- f) Jumlah kartu menyesuaikan dengan level permainan, semakin tinggi level semakin banyak kartu.
- g) *Depth First Search* sebagai pencarian solusi dan *Multiplicative CRNG* sebagai pembangkit permainan *Pairs*
- h) Permainan *Pairs* diimplementasikan pada *Android OS Mobile*

1.4 Tujuan Penelitian

Tujuan dari penelitian ini adalah sebagai media pembantu pembelajaran pengenalan kosakata bahasa Arab melalui sebuah *game*/permainan interaktif

1.5 Manfaat Penelitian

Manfaat penelitian ini adalah mengenalkan kosakata bahasa Arab melalui permainan *Pairs* bahasa Arab sekaligus melatih daya ingat untuk membantu menghafal kosakata bahasa Arab.

1.6 Sistematika Penulisan

Dalam penulisan skripsi ini, secara keseluruhan terdiri dari lima bab yang masing-masing bab disusun dengan sistematika sebagai berikut:

BAB I PENDAHULUAN

Pada bab ini merupakan bab pendahuluan, yang di dalamnya memuat latar belakang masalah, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian dan sistematika pembahasan.

BAB II TINJAUAN PUSTAKA

Pada bab tinjauan pustaka menjelaskan teori yang berhubungan dengan permasalahan penelitian yang meliputi: (1) Permainan (2) Bahasa Arab (3) Permainan Pairs (4) *Depth First Search* (5) *Multiplicative CRNG (Congruential Random Number Generator)* (6) Platform Android

BAB III ANALISIS DAN PERANCANGAN SISTEM

Bab ini menjelaskan tentang pembuatan desain dan perancangan permainan Pairs Bahasa Arab Menggunakan Algoritma *Depth First Search (DFS)* Dan *Multiplicative CRNG* Sebagai Pencarian Solusi Dan Pembangkit Permainan yang meliputi metode penelitian yang digunakan, perancangan aplikasi, desain aplikasi yang akan digunakan nantinya.

BAB IV HASIL DAN PEMBAHASAN

Pada bab ini menjelaskan hasil yang dicapai dari perancangan sistem dan implementasi program, yaitu dengan melakukan pengujian sehingga dapat ditarik kesimpulan.

BAB V PENUTUP

Pada bab terakhir berisi kesimpulan berdasarkan hasil yang telah dicapai dari pembahasan. Serta berisi saran yang diharapkan sebagai bahan pertimbangan oleh pihak-pihak yang akan melakukan pengembangan terhadap program ini.



BAB II

TINJAUAN PUSTAKA

2.1 Permainan/Game

Dalam bahasa Indonesia “Game” berarti “Permainan”. Permainan yang dimaksud dalam game juga merujuk pada pengertian sebagai “kelincahan intelektual” (*intellectual playbility*). Sementara kata “game” bisa diartikan sebagai arena keputusan dan aksi permainannya. Ada target-target yang ingin dicapai pemainnya. Kelincahan intelektual, pada tingkat tertentu, merupakan ukuran sejauh mana game itu menarik untuk dimainkan secara maksimal. (Arix, 2011:5)

Saat ini perkembangan game di komputer sangat cepat. Para pengelola industri game berlomba-lomba untuk menciptakan game yang lebih nyata dan menarik untuk para pemainnya. Hal inilah yang membuat perkembangan games di komputer sangat cepat. Sehingga games bukan hanya sekedar permainan untuk mengisi waktu luang atau sekedar hobi. Melainkan sebuah cara untuk meningkatkan kreatifitas dan tingkat intelektual para penggunanya Pada saat bersamaan, game juga secara nyata mempertajam daya analisis para penggunanya untuk mengolah informasi dan mengambil keputusan cepat dan jitu. (Arix, 2011:5)

2.1.1 Sejarah Perminan

Dahulu istilah game identik dengan anak-anak selaku pemain. Yang ada dibenak kita adalah suatu kegiatan yang dilakukan oleh anak-anak yang menurut mereka itu dapat menyenangkan hati mereka. Segala bentuk kegiatan yang

memerlukan pemikiran, kelincahan intelektual dan pencapaian terhadap target tertentu dapat dikatakan sebagai game. (Arix, 2011:5)

Menurut David Fox dan Roman Verhosek (2002:1), sejarah permainan dimulai dari munculnya permainan *Go*, yang populer di negara-negara telah ada sejak tahun 2000 SM. Bahkan permainan mirip *Backgamon (Tabula, Nard)* dicatat dalam *script* Romawi kuno. Seiring berkembangnya zaman dan munculnya komputer, maka banyak permainan yang dikembangkan sehingga bisa dimainkan di dalam computer dan disukai banyak orang.

Game *arcade* yang dipelopori oleh *hits* seperti *Pac-Man*, berhasil merebut perhatian para jutaan gamer di dunia. Selain itu, *Console system* seperti *Magnavox Odyssey*, *Atari 2006*, *Mattel Intellivision* dan *ColecoVision* menciptakan game yang bisa dimainkan dirumah. Kemudian pada tahun 1985, Nintendo Entertainment System (NES) membuat heboh para penggemar game dengan grafik dan gameworlds yang luar biasa, seperti game *Super Mario Brothers*.

Pada akhirnya, *computer gaming* semakin diterima dan populer lewat *bestseller* seperti: *Doom*, *Quake* dan *Tomb Rider*. Clearly ultra realistic 3D merupakan kunci sukses game tersebut, yang membuat pemain merasa seolah-olah berada dalam realitas game tersebut.

2.1.2 Klasifikasi Permainan

Menurut David Fox dan Roman Verhosek (2002:9), *game* dapat diklasifikasikan sebagai berikut:

1. *Action Games*

Merupakan *games* yang melibatkan pergerakan actor yang sangat reaktif. Jenis *game* ini juga biasanya memiliki grafis yang cukup tinggi dengan suara yang memacu adrenalin pemain. Contohnya *Quake*, *Missile Command*, dan *Pong*

2. *Combat Game*

Merupakan *game* yang biasanya melibatkan dua karakter sebagai tokoh utama game. Dua karakter tersebut mencoba saling mengalahkan. Seringkali salah satu karakter mempunyai kekuatan khusus. Contohnya *Virtua Fighter*, *Street Fighter*, dan *Mortal Kombat*.

3. *Adventure Game*

Merupakan *game* yang melibatkan seorang karakter yang mempunyai misi menemukan sesuatu yang dunia yang baru. Biasanya misi dari permainan ini mengadopsi sebuah film/buku yang memiliki cerita, karakter dan plot begitu kuat. Contohnya *Zork*, *Monkey's Island*, dan *Riven*

4. *Puzzle Game*

Permainan ini membutuhkan logika dari pemaian untuk memenangi permainan. Game ini membutuhkan ketelitian dan konsentrasi yang tinggi. Tetris merupakan salah satu jenis *game* yang masuk ke dalam jenis *puzzle game*.

5. *Strategy Game*

Merupakan permainan yang melibatkan banyak sekali kemungkinan. Biasanya membutuhkan strategi, ketelitian dan kecepatan dalam menyelesaikan

permainan jenis ini. Contoh dari *game* jenis ini adalah *Command and Conquer* and *Warcraft*

6. *Card Game*

Permainan jenis ini biasanya menggabungkan antara kesempatan dan ketrampilan dari pemain. Peraturan juga diterapkan dalam *game* jenis ini. Permainan *Pairs* merupakan salah satu permainan jenis ini.

2.2 Bahasa Arab

Bahasa merupakan alat komunikasi dan pendukung dalam pergaulan manusia dalam sehari-hari baik antara individu dengan individu, individu dengan masyarakat, maupun dengan bangsa tertentu. (Muhammad Ali Khauli, 1993:15)

Adapun bahasa Arab merupakan bahasa yang istimewa di dunia ini seperti yang kita ketahui, bahwasannya bahasa Arab tidak hanya merupakan bahasa peradaban, melainkan juga sebagai bahasa persatuan umat Islam di dunia. Bahasa Arab adalah selain merupakan bahasa Al-Qur'an (firman Allah atau kitab pedoman umat Islam) yang memiliki uslub yang bermutu juga memiliki sastra yang sangat mengagungkan manusia dan manusia tidak mampu untuk menandingi. Bahwa bahasa Arab merupakan bahasa orang Arab dan sekaligus juga merupakan bahasa Islam (Bambang Kiswanto, 2009:1)

Di dalam kitab suci Al Quran, Allah SWT secara khusus meletakkan keutamaan bahasa Arab :

إِنَّا أَنْزَلْنَاهُ قُرْآنًا عَرَبِيًّا لَعَلَّكُمْ تَعْقِلُونَ

Artinya:

Sesungguhnya kami menurunkannya berupa Al Quran dengan berbahasa Arab, agar kamu memahaminya.(QS. Yusuf/ 12:2)

Bahasa Arab adalah bahasa pemersatu bangsa Arab dan umat Islam yang telah memberikan kontribusi yang besar dalam memperkaya bahasa lain

termasuk bahasa Indonesia di mana banyak ditemukan kata-kata serapan yang berasal dari bahasa Arab. (Rahmatsyah Putra, 2010:1)

Menurut Arsyad (2004) dalam Rahmatsyah Putra (2010:1), Bahasa Arab merupakan salah satu bahasa dari sekian banyak bahasa di dunia ini dan merupakan salah satu bahasa mayor yang dituturkan oleh lebih dari 200.000.000 umat manusia. Bahasa ini digunakan secara resmi oleh kurang lebih 20 negara. Karena ia merupakan bahasa kitab suci dan tuntunan agama umat Islam sedunia, maka tentu saja ia merupakan bahasa yang paling besar signifikansinya bagi ratusan juta muslim sedunia, baik yang berkebangsaan Arab maupun bukan.

Begitu pentingnya kita mempelajari bahasa Arab juga tercantum dalam sebuah hadits nabi yang diriwayatkan oleh Thabrani yang artinya:

“Cintailah Bahasa Arab karena tiga hal, yaitu bahwa aku adalah orang Arab, bahwa Al Qur’an berbahasa Arab, dan bahasa penghuni syurga adalah bahasa Arab” (HR. Thabrani)

2.3 Permaianan Pairs

Menurut Erik Alfthan (2007:1) *Game Pairs* adalah sebuah *game* yang mempunyai alur di mana dalam sebuah tampilan utama ketika pemain memulai *game* akan muncul banyak gambar yang masih tersembunyi, di mana gambar-gambar tadi tiap-tiap gambarnya mempunyai sebuah pasangan yang gambarnya serupa. Tata letak gambar diacak agar pemain terkecoh dalam mencocokkan gambar-gambar tersebut, jika pemain tersebut mengklik sebuah gambar maka gambar itu akan terbuka.

Tugas dari pemain adalah membuka gambar yang serupa dengan gambar yang dia buka sebelumnya, jika gambar tersebut tidak cocok atau tidak

sama dengan gambar sebelumnya, maka gambar akan tertutup kembali dan pemain akan memilih gambar kembali, namun jika gambar yang dibuka antara gambar pertama dengan gambar yang dibuka selanjutnya cocok atau sama, maka gambar tersebut akan hilang terbuka. Permainan akan selesai jika pemain menemukan semua pasangan gambar



Gambar 2.1 Salah satu tampilan permainan Pairs

2.4 Depth First Search

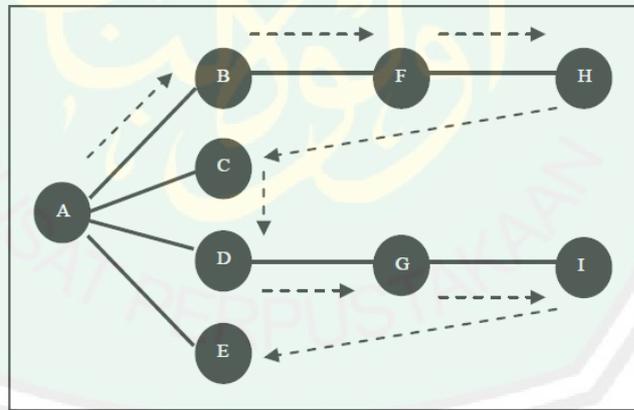
Algoritma *DFS* pertama kali diperkenalkan oleh Tarjan dan Hopcroft 20 tahun yang lalu. Mereka menunjukkan bagaimana *DFS* dapat digunakan untuk membangun sejumlah algoritma graf yang efisien. (Teneng dkk, 2010:58)

Algoritma *Depth First Search (DFS)* atau dengan nama lain algoritma pencarian mendalam adalah teknik penelusuran graf yang mirip dengan algoritma *Breadth First Search (BFS)*. Namun algoritma *DFS* melakukan penelusuran dengan mengunjungi secara rekursif mendalam salah satu dari sejumlah simpul yang dibangkitkan sebelumnya akhirnya pindah ke anak yang lain dari simpul akarnya. (Ali Akbar Septiandri, 2011:1)

Menurut Adi Nugroho, pendekatan *DFS* biasanya diimplementasikan menggunakan tumpukan (*Stack*), sementara pendekatan *BFS* biasanya diimplementasikan menggunakan antrian (*Queue*). (2008:450)

Menurut Robert Laforce (1998) dalam Adi (499:2008) menyatakan algoritma *Depth First Search (DFS)* memiliki beberapa aturan, yaitu:

1. Jika mungkin, lakukan kunjungan pada simpul-simpul pendamping (*adjacent vertex*) yang belum pernah dikunjungi, tandai dan masukkanlah (*push*) ke *stack*. Sebagai contoh, pada gambar di bawah ini, aturan ini dapat diterapkan pada simpul-simpul sebelum H. Pada langkah terakhir ini kita perlu melakukan sesuatu karena tidak ada simpul pendamping yang belum dikunjungi. Dalam hal ini kita bisa masuk ke aturan 2.



Gambar 2.2 Alur Algoritma *Depth First Search*

Tabel 2.1 Isi *Stack* dalam *DFS*

Event	Isi Stack
Visit A	A
Visit B	AB
Visit F	ABF
Visit H	ABFH
Pop H	ABF
Pop F	AB
Pop B	A
Visit C	AC
Pop C	A
Visit D	AD
Visit G	ADG
Visit I	ADGI
Pop I	ADG
Pop G	AD
Pop D	A
Visit E	AE
Pop E	A
Pop A	-
FINISH	

2. Jika saat kita melakukan aturan di atas kita mengalami kesulitan, maka kita keluarkan (*popped off*) simpul dari *stack*. Mengikuti aturan ini, jika kita mengeluarkan suatu simpul dari suatu *stack*, kita akan sampai simpul di bawahnya. Jika simpul di bawahnya bukan simpul pendamping yang belum dikunjungi, kita keluarkan lagi. Demikian selanjutnya hingga kita tidak bisa melakukannya lagi dan kita harus masuk ke aturan ketiga di bawah ini.

3. Jika kita tidak bisa mengikuti aturan 1 maupun 2 di atas, berarti kita telah menyelesaikan algoritma *Depth First Search*

Algoritma *Depth First Search* memiliki kelebihan dibandingkan dengan algoritma sejenisnya seperti *Breadth First Search* dalam persoalan solusi yang ditemukan, algoritma *Depth First Search* juga mempunyai kelebihan dalam hal pemakaian memori, dari pada algoritma *Breadth First Search*, karena *Breadth First Search* harus menyimpan semua simpul yang pernah dibangkitkan. Jika solusi yang dicari berada pada level yang dalam dan paling kiri, maka akan ditemukan solusi secara cepat.

Selain memiliki kelebihan di atas, algoritma ini juga memiliki kelemahan, antara lain jika pohon yang dibangkitkan mempunyai level yang dalam (tak terhingga).

2.5 Multiplicative CRNG (Congruential Random Number Generator)

Bilangan acak adalah bilangan sembarang namun tidak sembarangan. Bilangan acak tidak dapat diprediksi. Tidak ada bilangan acak yang benar-benar menghasilkan deret bilangan acak secara sempurna. Bilangan acak

yang dihasilkan dengan rumus-rumus matematika adalah bilangan acak semu (*pseudo*).

Menurut Kakiay (2004, 21), *Random Number Generator* adalah suatu algoritma yang digunakan untuk menghasilkan urutan-urutan atau *sequence* dari angka-angka sebagai hasil dari perhitungan dengan computer yang diketahui distribusinya sehingga angka-angka tersebut muncul secara *random* dan digunakan secara terus menerus

Menurut (Riani L, 2010:2), pembangkit bilangan acak itu harus:

- Berdistribusi uniform (0,1) dan tidak berkorelasi antar bilangan
- Membangkitkan secara cepat dan *storage* tidak besar
- Dapat di-“*reproduce*”
- Periode besar, karena kemungkinan bilangan acak dibangkitkan berulang

Ada banyak algoritma/metode untuk membangkitkan bilangan acak., salah satunya adalah *Multiplicative CRNG* (*Congruential Random Number Generator*). Algoritma/metode ini merupakan salah satu jenis dari algoritma *RNG* (*Random Number Generator*) yang dibuat oleh usulan Lehmer (1951).

Algoritma/metode *RNG* dituliskan sebagai berikut (Riani L, 2010:6):

Pseudo RNG, berbentuk:

$$Z_i = (aZ_{i-1} + c) \text{ mod } m$$

Dimana:

Z_i = bilangan acak ke- i dari deretnya

Z_{i-1} = bilangan acak sebelumnya

a = factor pengali

c = *increment*

m = *modulus*

Kunci pembangkit adalah Z_0 yang disebut **umpan** (*seed*)

Menurut Riani L. (2010:21), apabila nilai *increment* (c) adalah 0, maka disebut algoritma tersebut *Multiplicative Congruential Generator*. Maka bentuk *Pseudo RNG* dari *Multiplicative Congruential Generator* sebagai berikut:

$$Z_i = (aZ_{i-1}) \bmod m$$

Jika diimplementasikan ke dalam bahasa pemrograman java maka hasilnya adalah seperti di bawah ini:

```
public class mcrng {
    public int rand(int range, int zo, int a, int c,
int m) {
        for (int i = 0; i < range; i++) {
            zo = (a * zo + c) % m;
            System.out.print(zo + " ");
        }
        return zo;
    }
}
class main {
    static mcrng n = new mcrng();
    public static void main(String[] args) {
        n.rand(20, 2, 11, 0, 29);
    }
}
```

Output:

22 10 23 21 28 18 24 3 4 15 20 17 13 27 7 19 6 8 1 11

Dari percobaan di atas menunjukkan bahwa bilangan yang dibangkitkan menggunakan metode *Multiplicative CRNG* sebanyak 20 bilangan cukup bervariasi, dan tidak ada satupun bilangan yang mengalami perulangan.

2.6 Platform Android

Android merupakan sebuah *platform* untuk perangkat bergerak (*mobile device*) yang semakin populer. Bahkan, beberapa perusahaan riset telah menobatkan Android sebagai jawara ponsel pintar (*smartphone*) melebihi *platform* yang lain, seperti Symbian atau Apple iOS. (Eueung Mulyana, 2012:1)

Menurut Eueung Mulyana (2012:2) bahwa sebagai sebuah *platform*, Android adalah susunan dari beberapa perangkat lunak (*software stack*). Stack ini secara umum meliputi sistem operasi, *middleware* dan aplikasi-aplikasi kunci.

Android menggunakan Linux *kernel* yang dimodifikasi, walaupun begitu Android tidak bisa dikatakan sama dengan system Linux secara utuh karena beberapa komponen kunci Linux tidak disertakan.

Menurut Nazzaruddin Safaat (2012:2) bahwa tidak hanya menjadi system operasi di *smartphone*, saat ini Android menjadi pesaing utama dari Apple pada system operasi *Tablet PC*. Hal tersebut dikarenakan Android adalah *platform* yang sangat lengkap baik itu system operasinya, aplikasi dan *tool* pengembangan, *Market* aplikasi Android serta dukungan yang sangat tinggi dari komunitas *open source* dunia, sehingga Android terus berkembang pesat baik dari segi teknologi maupun dari segi jumlah *device* yang ada di dunia.

Beberapa keunggulan *Platform* Android adalah sebagai berikut (Nazzaruddin Safaat, 2012:3):

- 1) Lengkap (*Complete Platform*). Para desainer dapat melakukan pendekatan yang komprehensif ketika sedang mengembangkan *platform* Android. Android menyediakan banyak *tools* dalam membangun *software* dan merupakan

sistem operasi yang aman dan banyak menyediakan *tools* dalam membangun *software* dan memungkinkan untuk peluang pengembangan aplikasi

2) Terbuka (*Open Source Platform*). *Platform* Android disediakan melalui lisensi *open source*.

3) Bebas (*Free Platform*). Android merupakan *platform* atau aplikasi yang bebas untuk dikembangkan. Tidak ada lisensi atau biaya royalti untuk dikembangkan pada *platform* Android.



BAB III

ANALISA DAN PERANCANGAN

3.1 Analisa dan Perancangan Sistem

3.1.1 Keterangan Umum

Penulis membangun permainan Pairs bahasa Arab pada perangkat bergerak (*mobile device*), khususnya dalam perangkat bergerak dengan system operasi Android, penulis mengharapkan pemain dapat memainkan permainan Pairs dimanapun dan kapanpun. Karena peneliti sebelumnya membuat permainan Pairs pada sistem operasi yang berbasis *desktop*. Permainan Pairs berguna untuk mengasah daya ingat dari pemain. Penulis mengembangkan penelitian ini dengan menyisipkan unsur edukasi di dalamnya, yaitu dengan menambahkan tulisan dan suara kosakata bahasa Arab pada kartu yang dibuka oleh pemain. Pemain akan dikenalkan total sebanyak 18 kosakata bahasa Arab per levelnya berikut cara pelafalannya.

Permainan Pairs bahasa Arab terbagi menjadi beberapa level permainan, yaitu mudah, sedang dan sulit. Penentuan level permainan berdasarkan banyaknya jumlah kartu. Semakin sulit level permainan maka semakin banyak kartu yang akan tampil. Jenis gambar dari kartu yang keluar pada setiap levelnya tampil secara acak.

Jika pemain mengklik salah satu kartu yang tertutup maka permainan akan menampilkan *messege* tulisan kosakata dalam bahasa Arab dan mengeluarkan suara berdasarkan nama gambar yang terbuka, sehingga permainan

ini akan menjadi lebih interaktif, lebih menarik dan tentunya target yang penulis inginkan bisa tercapai yaitu membantu pemain dalam mengenal bahasa Arab, serta bagaimana mengucapkannya dengan yang benar.

Objek dalam penelitian ini, yaitu menerapkan algoritma *Depth First Search* untuk pencarian solusi permainan dan algoritma *Multiplicative CRNG* untuk membangkitkan permainan.

3.1.2 Rancangan Kategori Permainan

Permainan terdiri dari 4 kategori, yaitu *Amusing Animal*, *At Home*, *At School*, *Count With Me*. Pemilihan 4 kategori tersebut diambil dari sebuah situs kamus *online* bahasa Arab yg diterbitkan oleh Oxford University Press. Penulis memilih kamus online karena di dalam website tersebut sudah terdapat banak sekali kosakata bahasa Arab yang terkategori menurut jenisnya masing-masing, sehingga memudahkan dalam belajar bahasa Arab. Selain itu dalam website tersebut sudah dilengkapi dengan fitur audio dan gambar sehingga lebih interaktif.

Alamat dari situs tersebut adalah

<http://www.oup.com/oxed/children/firstwords/arabic>.



Gambar 3.1 Tampilan website kamus online Oxford University

Gambar di atas adalah tampilan dari situs kamus *online* terbitan dari Oxford University Press. Berdasarkan situs tersebut, penulis memasukkan 4 kategori dari sekian kategori yang ada dalam situs tersebut, karena menurut penulis 4 kategori yang dipilih sudah mewakili kosakata-kosakata yang umum dan sering diucapkan oleh anak-anak. Setiap kategori terdapat 18 gambar yang muncul secara acak dan terbagi menjadi 3 level permainan.

Tabel 3.1 Kategori Permainan

Amusing Animal	At Home	At School	Count With Me
Mouse = فأر	Mirror = مرآة	Our Classroom = فصلنا	0 = صفر
Elephant = فيل	Tree = شجرة	Schoolbag = حقيبة مدرسية	1 = واحد
Crocodile = تمساح	Door = باب	Ruler = مسطرة	2 = اثنان
Giraffe = زرافة	Window = نافذة	Lunch Box = عربة الغداء	3 = ثلاثة
Parrot = ببغاء	Leaf = ورقة (نبات)	Book = كتاب	4 = اربعة
Penguin = بطريق	Flower = زهرة	Computer = حاسوب	5 = خمسة
Fish = سمكة	Garage = مرآب	Blackboard = سبورة	6 = ستة
Tiger = نمر	Lamp = مصباح	Map = خريطة	7 = سبعة
Hippopotamus = فرس النهر	Fence = سياج	Chalk = طباشير	8 = ثمانية
Zebra = حمار وحشي	Letters = رسائل	Globe = الكرة الأرضية	9 = تسعة
Kangaroo = كنغر	Snail = قوقع	Bicycle = دراجة	10 = عشرة
Panda = البندا	Spider = عنكبوت	Desk = مكتب	11 = إحدى عشرة
Monkey = قرد	Stairs = سلم	Keyboard = لوحة مفاتيح	12 = إثني عشرة
Koala = كوال	Gate = بوابة		13 = ثلاثة عشرة
Rhinoceros = وحيد القرن	Trash can = سلة المهملات		14 = أربعة عشرة
			15 = خمسة عشرة

Cat = قطة	Roof = سقف	Magnet = مغناطيس	16 ستة عشرة =
Dog = كلب	Dog = كلب	Mouse = فأرة	17 سبعة عشرة =
Rabbit = أرنب	Cat = قطة	Trash can = سلة المهملات =	
		School = مدرسة	
		Chair = كرسي	

Tabel di atas adalah rincian pengelompokan kosakata bahasa Arab yang dibagi menjadi 4 kategori. Kosakata-kosakata tersebut adalah kosakata yang umum yang berhubungan langsung anak-anak. Kategori-kategori tersebut akan tampil dalam bentuk gambar dan juga suara. Penulis mengharapkan pemain akan lebih mudah dalam belajar bahasa Arab.

3.1.3 Rancangan Level Permainan

Permainan terdiri dari 3 level, yaitu mudah, sedang dan sulit. Level permainan semakin tinggi maka semakin banyak kartu yang akan tampil dan harus pemain harus menemukan pasangannya. Berikut adalah tabel pembagian level permainan:

Tabel 3.2 Jumlah kartu yang ditampilkan setiap level

Mudah	Sedang	Sulit
16 kartu	30 kartu	36 kartu

Level mudah menampilkan 16 kartu atau ada 8 pasang gambar, sedangkan pada level sedang menampilkan 30 kartu atau ada 15 pasang gambar, dan untuk level sulit kartu menampilkan 36 kartu atau ada 18 pasang gambar.

Penampilan gambar dari setiap level akan meningkat, contohnya semua kartu yang tampil pada level mudah, juga akan tampil di level sedang dan sulit,

dan semua kartu yang tampil di level sedang juga akan tampil di level sulit, sehingga gambar yang tampil pada level sulit adalah kumpulan gambar pada level mudah dan sedang ditambah dengan sisa gambar yang memang hanya tampil pada level sulit saja. Penampilan jenis gambar berdasarkan kategori yang sudah dipilih sebelumnya.

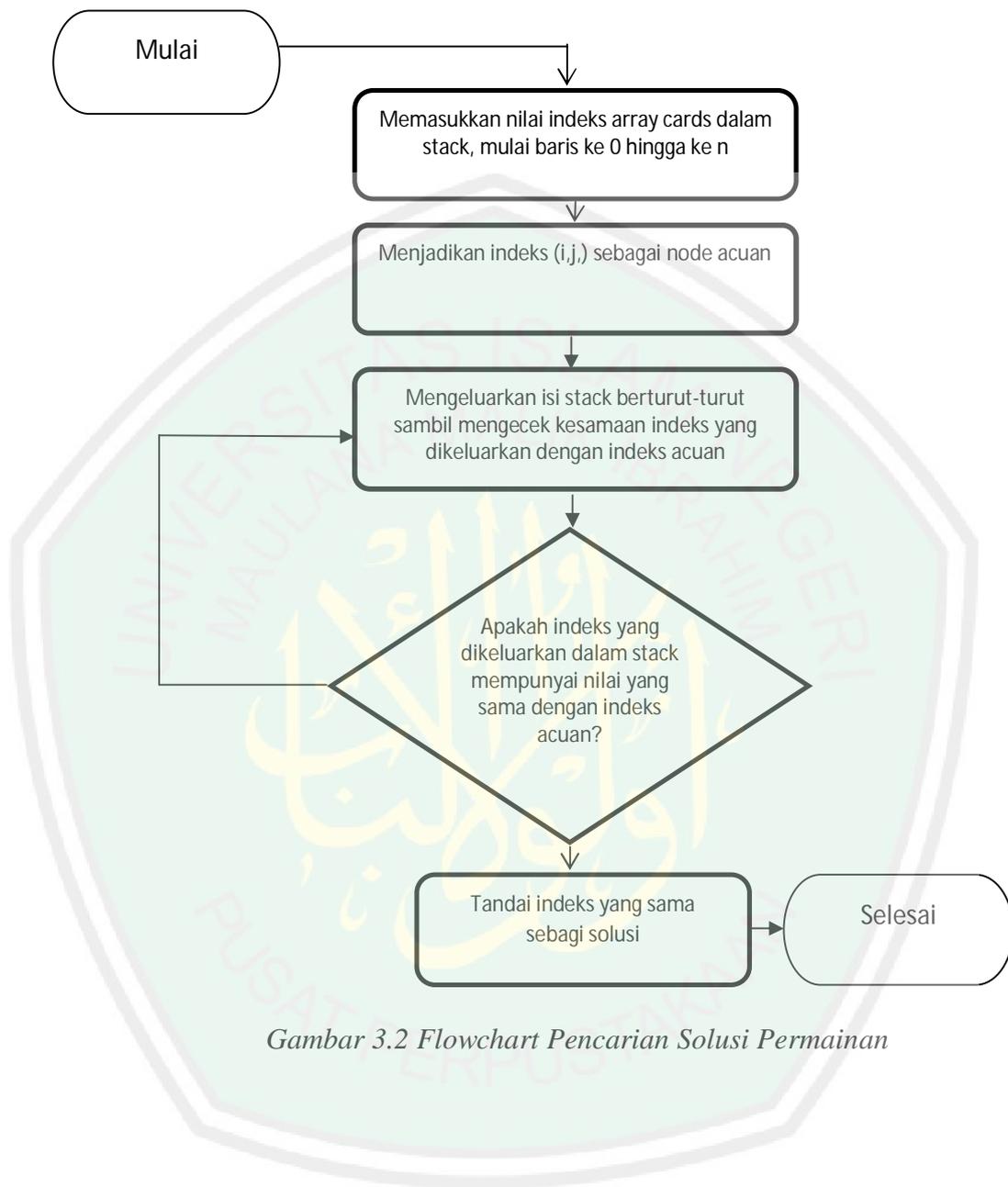
Perbedaan level permainan berdasarkan jumlah kosakata yang ditampilkan. Banyaknya kosakata yang tampil menunjukkan tingkat kesulitan level dari permainan. Permainan akan menampilkan kosakata-kosakata berupa kartu bergambar dan suara yang sesuai dengan gambar jika pemain mengklik pada salah satu kartu, jika pemain menemukan pasangan gambar maka akan muncul tulisan dalam bahasa Arabnya beserta bahasa Inggrisnya.

3.2 Rancangan Algoritma Depth First Search dan Multiplicative CRNG

Permainan ini akan mengimplementasikan algoritma *Depth First Search* sebagai pencarian solusi permainan dan algoritma *Multiplicative CRNG* sebagai pembangkit permainan.

3.2.1 Pencarian Solusi Permainan

Penulis menerapkan algoritma *DFS* untuk menemukan solusi dalam permainan, yaitu ketika pemain merasa kesulitan dalam menemukan pasangan kartu kemudian system akan melakukan pencarian pasangan kartu secara otomatis.



Gambar 3.2 Flowchart Pencarian Solusi Permainan

	0	1	2
0	1	2	3
1	4	5	6
2	1	2	3
3	4	5	6

Gambar 3.3 Bentuk matriks permainan Pairs

Tabel 3.3 Alur Penelusuran Pencarian Solusi

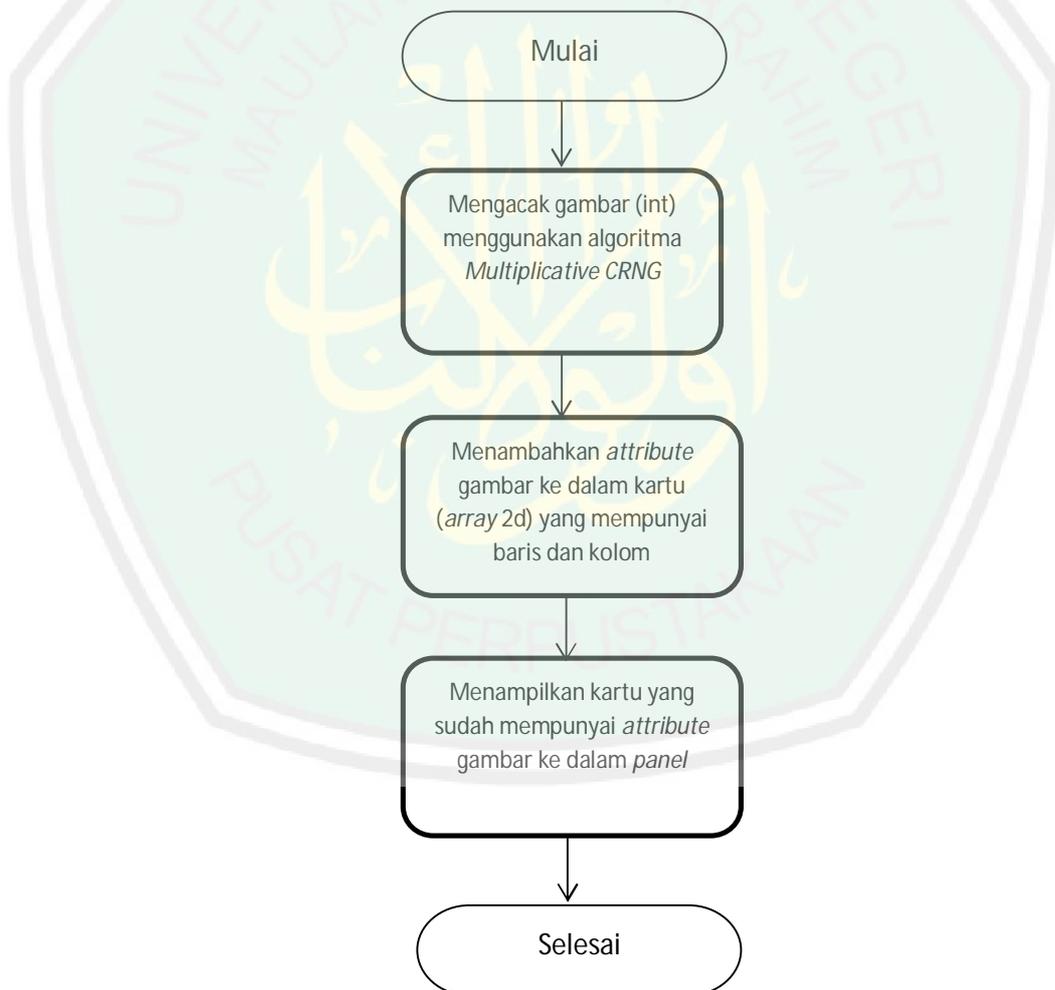
no	Event	Isi Stack	no	Event	Isi Stack
1	Push (0,0)	1	13	Push (2,0)	1
2	Push (0,1)	1,2	14	Push (2,1)	1,2
3	Push (0,2)	1,2,3	15	Push (2,2)	1,2,3
4	Pop (0,2)	1,2	16	Pop (2,2)	1,2
5	Pop (0,1)	1	17	Pop (2,1)	1
6	Pop (0,0)	-	18	Pop (2,0)	-
7	Push (1,0)	4	19	Push (3,0)	4
8	Push (1,1)	4,5	20	Push (3,1)	4,5
9	Push (1,2)	4,5,6	21	Push (3,2)	4,5,6
10	Pop (1,2)	4,5	22	Pop (3,2)	4,5
11	Pop (1,1)	4	23	Pop (3,1)	4
12	Pop (1,0)	-	24	Pop (3,0)	-
FINISH					

Gambar dan tabel di atas merupakan gambaran permainan Pairs. Permainan Pairs ditampilkan dalam bentuk array 2 dimensi. Setiap indeks mempunyai nilai integer. Setiap indeks memiliki pasangan, yaitu yang mempunyai nilai integer sama.

Pencarian solusi dilakukan menggunakan algoritma DFS yang biasanya dalam proses penelusuran node menggunakan stack. Pertama kali indeks (0,0) dijadikan sebagai indeks acuan untuk dicari kesamaannya. Penelusuran pertama dimulai dengan mengunjungi node/indeks pada baris pertama. Nilai-nilai pada indeks tersebut di push/dimasukkan ke dalam stack secara berurutan mulai dari indeks yang paling kiri sebanyak jumlah kolom. Setelah itu nilai pada indeks yang terakhir kali masuk ke dalam stack langsung di pop/dikeluarkan sambil dicek kesamaannya dengan nilai integer pada indeks acuan (0,0). Proses pengeluaran nilai dari stack berurutan mulai dari nilai pada indeks yang terakhir kali masuk hingga nilai pada indeks yang masuk pertama kali dalam stack. Sering proses ini kita sebut dengan Last In First Out (LIFO). Jika nilai pada indeks yang dikeluarkan sama maka indeks tersebut ditandai sebagai solusi. Jika tidak sama maka penelusuran akan dilanjutkan dengan indeks pada baris berikutnya dengan kolom ke 0 yaitu (1,0). Begitu seterusnya hingga ditemukan solusi indeks yang mempunyai nilai integer yang sama dengan indeks acuan. Node/indeks yang sudah ditemukan sebagai solusi tidak akan dilakukan pengecekan lagi jika indeks acuan sudah berubah.

3.2.2 Pembangkit Permainan

- a. Mengacak indeks *array* gambar menggunakan algoritma *Multiplicative CRNG*
- b. Mengisi *array* 2 dimensi (kartu) yang mempunyai indeks posisi dengan *attribute integer* dari *array* gambar.
- c. Menampillkan kartu yang sudah mempunyai *attribute integer* gambar ke dalam *panel*.



Gambar 3.4 Flowchart Pembangkit Permainan

3.3 Perancangan Aplikasi

Untuk mempermudah mengingat aplikasi ini, aplikasi permainan Pairs bahasa Arab ini diberi nama **Arabic Pairs**.

3.3.1 Antarmuka Permainan

Dalam Permainan Arabic Pairs terdapat beberapa halaman sebagai berikut:

a) Halaman Utama



Gambar 3.5 Rancangan Halaman Utama

Di halaman utama terdapat beberapa tombol antara lain:

- New Game, tombol untuk memulai permainan
- Score, tombol untuk melihat skor
- Help, tombol untuk melihat petunjuk permainan
- Exit, tombol untuk keluar dari aplikasi
- , tombol untuk mengaktifkan/menonaktifkan audio

b) Halaman Pilih Kategori Score

Score
Easy
Medium
Hard

Gambar 3.6 Rancangan Halaman Pilih Skor

Halaman ini muncul setelah pemain memilih menu **Score** pada halaman utama. Pada halaman ini pemain bisa melihat skor pada tiap level permainan, mulai dari level *Easy*, *Medium*, dan *Hard*

c) Halaman Score

Player	Clicks	Date
Aji	18	1 Mar
Wahyu	24	2 Mar

Gambar 3.7 Rancangan Halaman Skor

Gambar di atas merupakan rancangan halaman skor. Halaman tersebut berisi info skor yang terdiri dari nama pemain, jumlah klik dan tanggal main.

d) Halaman Help

Pada halaman ini akan ditampilkan cara memainkan permainan ini.

e) Halaman Pilih Kategori



Category
Amusing Animal
At Home
At School
Count With Me

Gambar 3.8 Rancangan Halaman Pilih Kategori

Halaman ini muncul setelah pemain memilih menu New Game pada halaman utama. Pada halaman ini pemain memilih kategori permainan yang diinginkan. Terdapat 4 kategori yang bisa dipilih oleh pemain, yaitu *Amusing Animal* (Hewan-hewan), *At home* (Di Rumah), *At school* (Di Sekolah), dan *Count With Me* (Angka-angka).

Halaman di atas merupakan halaman utama permainan Arabic Pairs. Halaman permainan ini sebenarnya masih satu halaman dengan halaman pilih level di atas. Halaman ini akan muncul beberapa kartu yang sudah tertata membentuk matriks 2 dimensi. Jumlah kartu sesuai dengan level permainan. Awal permainan semua kartu ditampilkan dalam kondisi tertutup. Permainan menampilkan kartu-kartu secara acak menggunakan algoritma *Multiplicative CRNG*. Setiap kartu pasti mempunyai pasangan. Tugas pemain adalah membuka kartu tersebut dan menemukan pasangannya. Cara membuka kartu tersebut adalah dengan mengklik pada salah satu kartu. Setiap kartu yang terpilih akan mengeluarkan suara berbahasa Arab sesuai dengan gambar yang terbuka. Permainan akan selesai jika pemain berhasil menemukan semua pasangan gambar yang ada.

Halaman ini juga terdapat tombol solusi. Fungsi dari tombol solusi ini adalah memberikan bantuan kepada pemain jika merasa kesulitan menemukan pasangan gambar. Jika pemain mengklik tombol ini maka sistem secara otomatis akan mencari 1 pasangan gambar menggunakan algoritma *Depth First Search*, sehingga pemain akan mengetahui posisi pasangan kartu yang mempunyai gambar yang sama. Sistem hanya akan memberikan 4 kali kesempatan untuk menggunakan tombol ini kepada pemain.

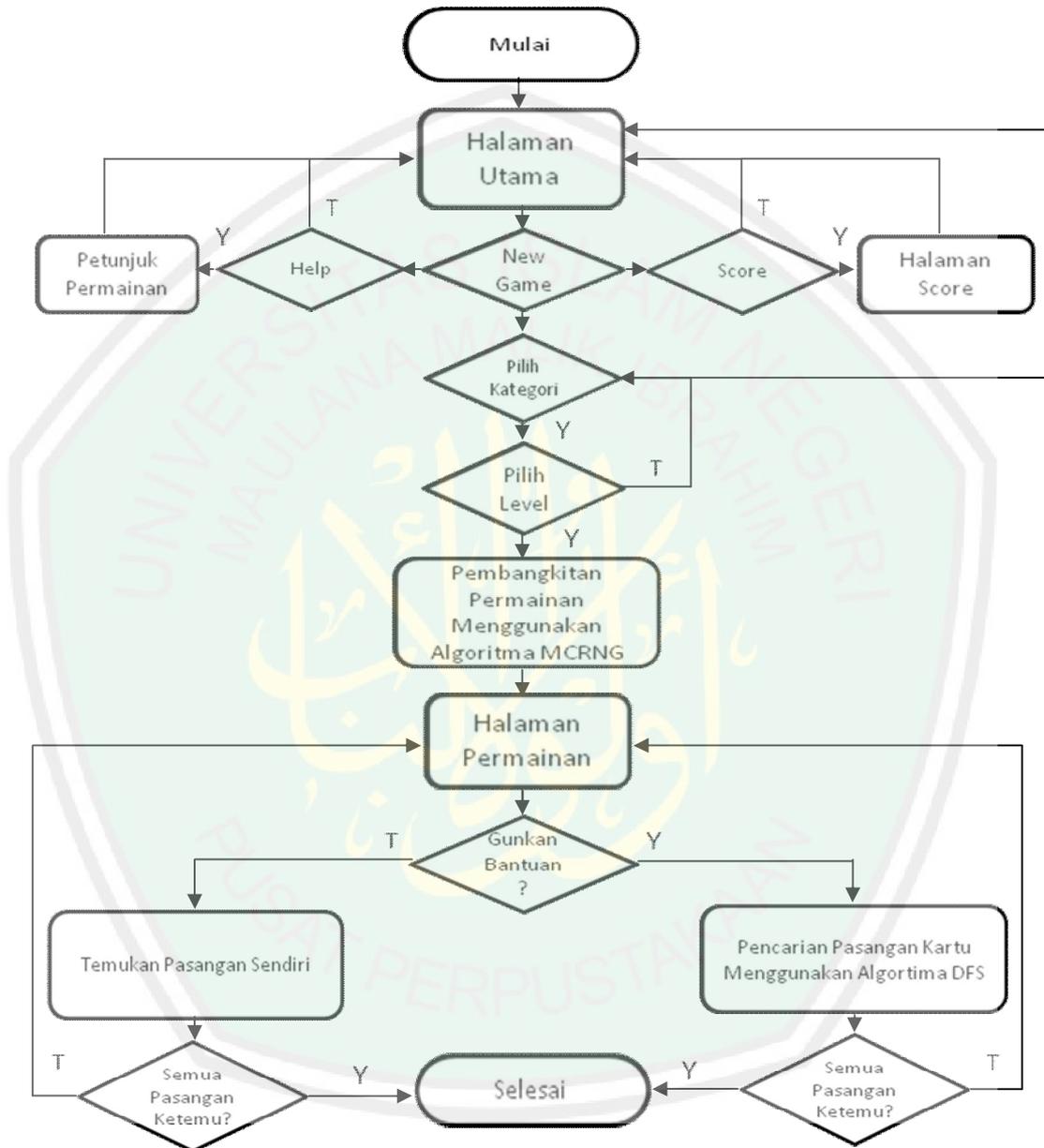
h) Halaman *Entry Player Name*

Congratulations
Player
Wahyu
Clicks 24 SAVE

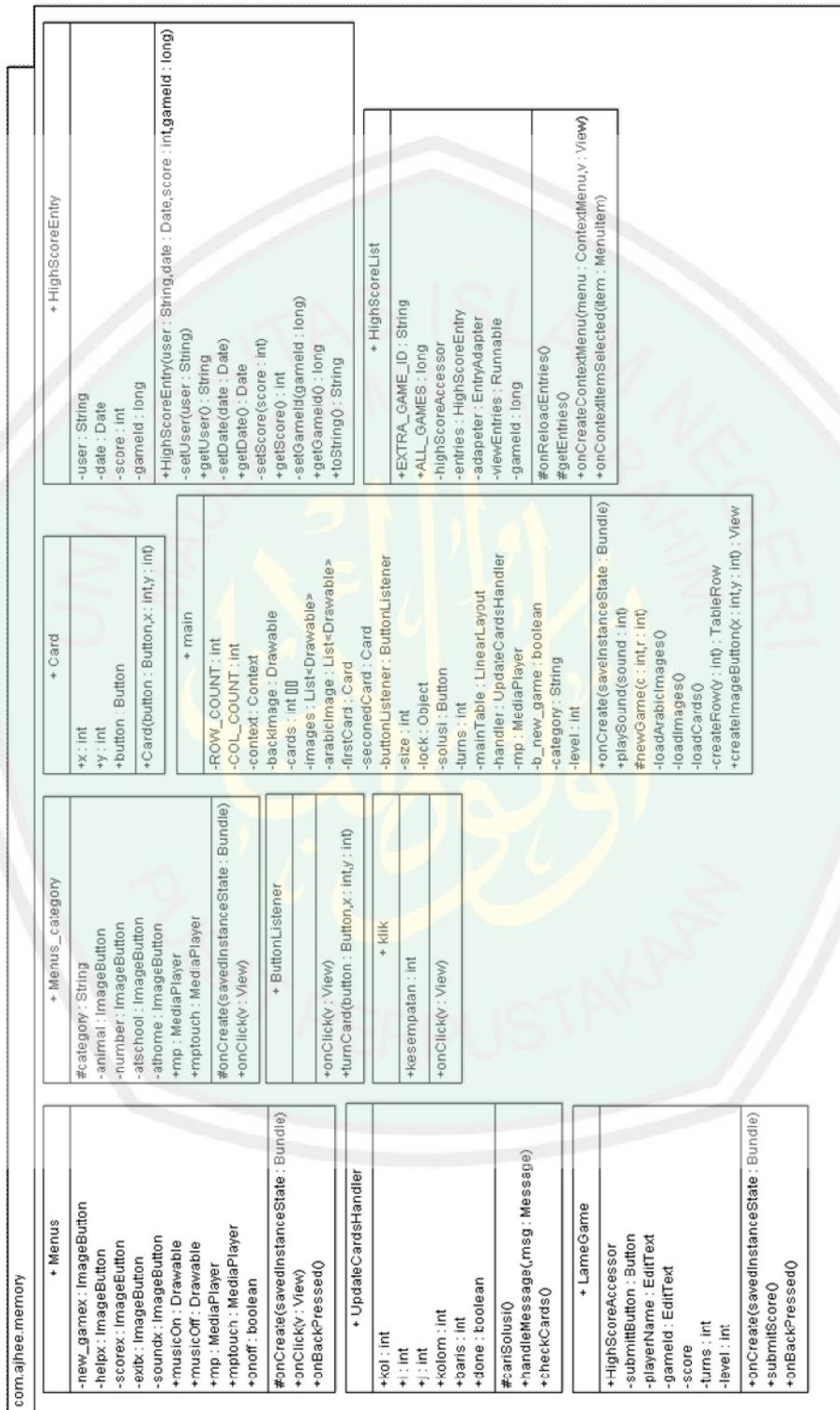
Gambar 3.11 Rancangan Halaman *Entry Player Name*

Halaman ini akan muncul setelah pemain berhasil menyelesaikan permainan, yaitu setelah pemain berhasil menemukan semua pasangan gambar. Skor dihitung berdasarkan seberapa banyak pemain melakukan klik selama proses menebak kartu yang sama. Semakin sedikit pemain melakukan klik maka posisi skor mereka akan semakin baik, begitu juga sebaliknya.

Berikut adalah alur dari permainan Arabic Pairs:



Gambar 3.12 Flowchart Permainan Arabic Pairs



Gambar 3.13 Class Diagram Permainan Arabic Pairs

3.3.2 Kebutuhan Sistem

Berikut ini adalah beberapa perangkat keras dan lunak yang dibutuhkan untuk mendukung pembuatan dan uji coba aplikasi Arabic Pairs.

a) Perangkat Keras (*Hardware*)

- PC/Laptop dengan spesifikasi minimal Processor Intel(R) Pentium(R) CPU P6200 @2.13 GHz, RAM 2GB dengan system operasi Windows XP SP 3 digunakan untuk pembuatan aplikasi
- *Smartphone* dengan spesifikasi minimal mempunyai sistem operasi Android versi 2.3 (Ginger Bread) digunakan untuk uji coba aplikasi

b) Perangkat Lunak (*Software*)

- Java, digunakan untuk dapat melakukan kompilasi aplikasi Android. Versi yang digunakan Sun Java SE versi 1.7 atau versi di atasnya.
- Software Eclipse, merupakan software IDE (*Integrated Development Environment*) yang digunakan untuk membangun aplikasi Android. Eclipse yang digunakan adalah versi Galileo yang sudah support dengan Android Development Tools (ADT)
- ADT (Android Development Tools), plugin tambahan untuk Eclipse yang dibutuhkan untuk membuat aplikasi Android. ADT yang digunakan adalah versi 8.0.1
- Android SDK (Software Development Kit), yang diperlukan sebagai alat bantu dan API dalam mengembangkan aplikasi Android menggunakan bahasa java. Android SDK yang digunakan adalah versi r8

BAB IV

HASIL DAN PEMBAHASAN

Pada bab ini membahas tentang hasil implementasi dari perancangan sebelumnya.

4.1 Implementasi Algoritma Multiplicative CRNG dan Depth First Search

4.1.1 Multiplicative CRNG

Algoritma *Multiplicative CRNG* diimplementasikan pada saat proses pengacakan nilai pada setiap posisi indeks kartu. Algoritma ini dimasukkan dalam sebuah method untuk membangkitkan kartu yang bernama *loadCards()*.

Method *loadCards()* dimasukkan rumus dari *Multiplicative CRNG* yaitu $Z_i = (aZ_{i-1} + c) \bmod m$. Variable yang dibutuhkan dari rumus tersebut antara lain **zo**, **a**, dan **m**, sedangkan variable **c** tidak dipakai karena selalu bernilai 0. Pendeklarasian variable tersebut seperti di bawah ini:

```
size = ROW_COUNT * COL_COUNT;

/*
 * Metode MCRNG inisialisasi variable a,zo,m,c
 */

int a = 0;
int zo = 0;
int aSulit[] = { 2, 5, 13, 15, 17, 18, 19, 20, 22, 24,
32, 35, 39,42, 50 };
int aSedang[] = { 3, 11, 12, 13, 17, 21, 22, 24, 34,
42, 43, 44, 48 };
int aMudah[] = { 3, 5, 6, 7, 10, 11, 12, 14, 20, 22,
23, 24, 27, 28, 29, 31, 37, 39, 40, 41, 44, 45, 46, 48 };
int zoA[] = { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12,
13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26,
27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40,
41, 42, 43, 44, 45, 46, 47, 48, 49, 50 };
int c = 0;
int m = size + 1;
```

Variable **a** merupakan sebuah nilai yang bertipe *array integer*. Variable **a** dibagi menjadi 3 macam, yaitu **aSulit**, **aSedang**, dan **aMudah**. Pembagian variable **a** menjadi 3 ini karena di dalam permainan ini terdapat 3 level yang setiap levelnya mempunyai jumlah kartu yang berbeda pula, jika variable **a** disamakan maka hasil pengacakan yang didapat nantinya akan tidak maksimal bahkan akan menghasilkan angka yang sama yang berulang-ulang.

Penentuan nilai variable **a** sendiri mulai dari **aSulit**, **aSedang**, dan **aMudah** sudah dilakukan proses seleksi dari ujicoba sendiri. Angka-angka dalam variable tersebut merupakan hasil pemilihan dari 50 angka mulai 1-50. Penulis melakukan pengecekan satu per satu mulai 1-50 di mana setiap angka yang menghasilkan pengacakan yang maksimal artinya tanpa ada pengulangan akan dimasukkan sebagai nilai variable **a**.

Variable **zo** diinisialisasikan sebagai variable **zoA** yang bertipe *array integer*. Nilai dari variable ini merupakan angka integer 1-50. Pemilihan angka tersebut didasarkan pada ketentuan dari rumus *Multiplicative CRNG* itu sendiri. Bahwa **z** hanyalah angka permulaan saja. Nilainya bisa sebarang, asal bilangan bulat. Penulis mengambil 50 angka pertama karena dirasa sudah cukup karena nantinya akan dikombinasikan dengan variable **a** yang diambil dari salah satu dari 3 jenis *array integer* yaitu **aSulit**, **aSedang**, dan **aMudah** menyesuaikan level yang dipilih pemain.

Selain dua variable di atas ada variable **size** dan **m**. Kedua variabel ini berfungsi untuk membatasi jumlah angka acak yang akan keluar, di mana **size** ini disesuaikan dengan level yang dipilih oleh pemain pada saat akan memulai

permainan. Nilai **size** ada tiga yaitu, 16, 30, dan 36. Variabel **m** bernilai **size** ditambahkan 1, artinya nilai **m** menyesuaikan dengan nilai **size**, yaitu 17, 31, dan 37. Variable **m** berfungsi sebagai nilai *modulus*.

Setelah pendeklarasian variable, masih di dalam method *loadCards()*, kemudian dilakukan fungsi percabangan *if*, untuk memilih variable-variabel di atas sesuai dengan level yang dipilih. Potongan *source code* nya adalah:

```

if (size == 36) {
  /* membangkitkan variable a dan zo level sulit */
  a = aSulit[(int) (Math.random() * aSulit.length)];
  zo = zoA[(int) (Math.random() * zoA.length)];
}
if (size == 30) {
  /* membangkitkan variable a dan zo level sedang */
  a = aSedang[(int) (Math.random() * aSedang.length)];
  zo = zoA[(int) (Math.random() * zoA.length)];
}
if (size == 16) {
  /* membangkitkan variable a dan zo level mudah */
  a = aMudah[(int) (Math.random() * aMudah.length)];
  zo = zoA[(int) (Math.random() * zoA.length)];
}

```

Potongan *source code* memperlihatkan bahwa untuk menentukan nilai variable **a** dan variable **zo** dibangkitkan secara acak berdasarkan posisi indeks dari masing-masing variable, diharapkan kemunculan variable **a** dan **zo** dapat muncul secara dinamis dengan kombinasi nilai yang bervariasi.

Setelah mengambil nilai variable **a** dan **zo** kemudian permainan melakukan pembangkitan permainan menggunakan algoritma *Multiplicative CRNG* dapat dilihat pada potongan *source code* di bawah ini:

```

// perulangan indeks i sebanyak size
for (int i = 0; i < size; i++) {
// membangkitkan bilangan acak zo
zo = (a * zo + c) % m;
// membentuk indeks posisi kartu dan memberi nilai pada
indeks kartu cards[][]
  cards[i % COL_COUNT][i / COL_COUNT] = zo % (size / 2);
}

```

Sistem mulai membangkitkan permainan melalui perulangan mulai dari indeks ke i yaitu 0, kemudian masuk ke dalam rumus dari algoritma *Multiplicative CRNG* untuk membangkitkan bilangan acak indeks pertama. Nilai-nilai dari variable **zo**, **a**, **c**, dan **m** menyesuaikan dengan pemilihan level permainan. Setelah menghasilkan bilangan acak pertama **zo**, kemudian masuk ke proses pemberian atribut nilai *integer* dengan rumus $zo \% (size / 2)$ dari posisi indeks kartu di mulai dari indeks ke (0,0). Atribut nilai *integer* yang dihasilkan tiap indeks ke (i,j) nantinya akan mempunyai nilai berpasang-pasangan, jika perulangan pembangkitan bilangan acak menggunakan algoritma *Multiplicative CRNG* sudah selesai dilakukan hingga indeks terakhir sesuai panjang **size**.

4.1.2 Depth First Search

Game ini menggunakan algoritma *Depth First Search* sebagai pencarian solusi permainan, di mana pada waktu pemain merasa kesulitan dalam mencari pasangan gambar maka pemain bisa menekan tombol **hint** maka secara otomatis system akan melakukan pencarian pasangan gambar dengan menggunakan algoritma *Depth First Search*.

Berikut adalah potongan *source code* algoritma *Depth First Search* yang telah berhasil diimplementasikan ke dalam bahasa pemrograman java.

```

protected void cariSolusi() {

int kolom = cards[0].length;
int baris = cards.length;
int a_baris = 0;
int a_kolom = 0;
int kol = a_kolom;
int bar = a_baris;

Stack<Integer> st = new Stack<Integer>();
boolean done = false;
while (!done) {
    a_baris = bar;
    a_kolom = kol;
    int acuan = cards[a_baris][a_kolom];
    if (acuan != 99) {
        for (int i = a_baris; i < baris; i++) {
            for (int j = a_kolom; j < kolom; j++) {
                st.push(cards[i][j]);
            }
            for (int p = kolom - 1; p >= a_kolom; p--){
                int cek = st.pop();
                if (cek == acuan);
                ((TextView) findViewById(R.id.tv1)).setText(i+"0"+ p);
                findViewById(R.id.tv1).setVisibility(View.VISIBLE);
                String temp = String.valueOf(((TextView)
                findViewById(R.id.tv1)).getText());
                int idB1 = Integer.parseInt(temp.substring(0,3));
                Animation animRotate =
                AnimationUtils.loadAnimation(context,
                R.anim.anim_alpha);
                findViewById(idB1).startAnimation(animRotate);
                done = true
            }
        }
        a_kolom = 0;
    }

    if (kol == kolom - 1) {
        bar++;
        kol = 0;
    } else {
        kol++;
    }
}
}

```

Potongan *source code* di atas merupakan *method* untuk pencarian solusi permainan atau mencari pasangan kartu secara otomatis. Proses awal dengan pendeklarasian *variable*. Proses berikutnya mengecek apakah nilai pada indeks **acuan** sudah terbuka/ditemukan solusi atau belum ditandai dengan nilai 99. Jika nilai tidak sama dengan 99 maka dilanjutkan pada proses memasukkan nilai pada indeks (i,j) pada baris 0 ke dalam *stack* ditandai dengan *method push*. Proses memasukkan nilai pada *stack* dilakukan sebanyak kolom yang ada pada baris tersebut. Setelah semua indeks sudah masuk ke dalam *stack*, nilai pada indeks yang terakhir kali masuk akan langsung dikeluarkan dari *stack* menggunakan *method pop* sekaligus dicek kesamaan nilainya satu per satu hingga kolom pertama. Jika nilai pada indeks tertentu sama dengan nilai pada indeks acuan, maka indeks tersebut ditandai sebagai solusi, pencarian selesai. Namun jika tidak ada nilai yang sama maka pencarian akan dilakukan pada indeks pada baris berikutnya dengan proses pengecekan sama dengan baris pertama di atas, hingga ada indeks yang mempunyai nilai yang sama dengan indeks acuan.

Jika pencarian menemukan solusi, kedua kartu pada indeks tersebut akan ditandai dengan animasi kartu berkedip-kedip. Sehingga pemain akan mengetahui kartu pada posisi mana yang mempunyai gambar yang sama. *Method cariSolusi* ini terletak pada tombol **hint** yang bisa digunakan oleh pemain sebagai bantuan mencari pasangan kartu secara otomatis. Pemain hanya akan dibatasi sebanyak 4 kali penggunaan.

4.2 Implementasi Aplikasi

Berikut adalah hasil implementasi aplikasi Arabic Pairs:



Gambar 4.1 Halaman Utama

Pada halaman utama terdapat 5 menu pilihan yaitu, **New Game**, **Score**, **Help**, **Exit** dan **Music**. Untuk melihat cara memainkan permainan ini pemain bisa mengklik tombol **Help**, maka akan muncul tampilan seperti dibawah in:



Gambar 4.2 Halaman Help

Apabila ingin memulai permainan baru pemain harus menekan tombol **New Game** kemudian menuju halaman pilih kategori sebagai berikut:



Gambar 4.3 Halaman Kategori

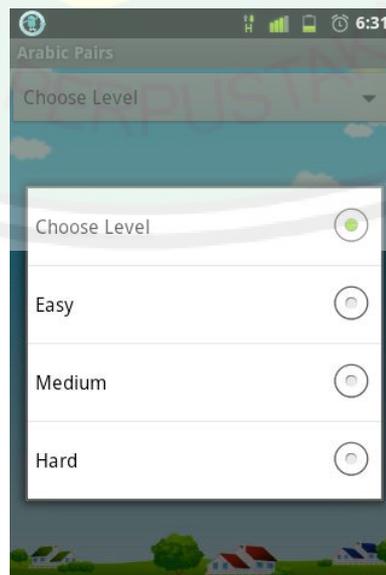
Pada halaman ini pemain bisa memilih kategori gambar yang ingin dimainkan. Ada 4 kategori yaitu *Amusing Animal* (Hewan), *At Home* (Di Rumah), *At School* (Di Sekolah), dan *Count With Me* (Angka-Angka).

Setelah memilih kategori kemudian akan muncul halaman pilih level seperti gambar di bawah ini:



Gambar 4.4 Halaman Pilih Level

Jika diklik **Choose Level** maka akan keluar 3 pilihan level permainan yaitu *Easy* (Mudah), *Medium* (Sedang) dan *Hard* (Sulit) seperti tampak pada gambar berikut:



Gambar 4.5 Pilihan Level Permainan

Jika pemain sudah memilih salah satu level permainan, maka muncul halaman permainan. Halaman ini adalah tempat permainan akan dimainkan. Berikut gambar halaman permainan:



Gambar 4.6 Halaman Permainan

Pada halaman ini terdapat *ImageButton* yang dibangkitkan dengan algoritma *Multiplicative CRNG* dengan gambar tanda tanya. Banyaknya *ImageButton* tergantung pada pemilihan level permainan. *ImageButton* tersebut terdapat mempunyai gambar-gambar tersembunyi yang tidak terlihat. Jenis Gambar yang tampil menyesuaikan dengan kategori yang dipilih.

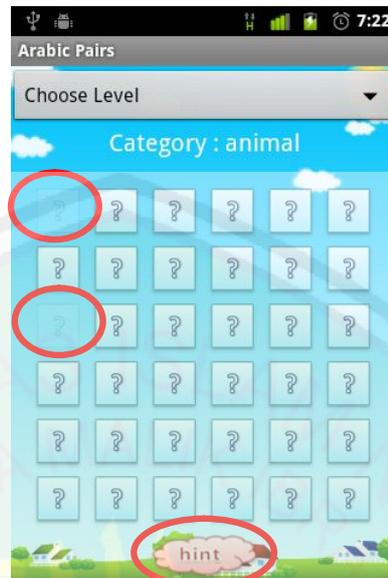
Pemain harus menemukan semua pasangan kartu yang tampil. Pemain cukup mengklik pada salah satu kartu bergambar tanda tanya untuk mengetahui gambar yang ada dibalik kartu tersebut, kemudian muncul gambar yang diikuti suara dari kartu yang telah dibuka. Gambar tampilan jika pemain membuka kartu sebagai berikut:



Gambar 4.7 Kartu Terbuka

Kartu terbuka setelah pemain mengklik pada salah satu kartu bergambar tanda tanya, jika kartu pertama dan kedua tidak sama maka kedua kartu tersebut akan tertutup kembali, dan pemain memilih kartu lain yang mempunyai gambar yang sama.

Halaman permainan ini juga terdapat tombol **hint** yang berfungsi sebagai bantuan kepada pemain jika merasa kesulitan dalam menemukan pasangan kartu menggunakan algoritma *Depth First Search*. Kesempatan menggunakan tombol **hint** ini dibatasi hanya 4 kali penggunaan. Tombol **hint** dapat digunakan ketika kartu tidak ada yang terbuka, artinya kartu dalam keadaan tertutup. Berikut gambar tampilan jika pemain menggunakan tombol **hint**:



Gambar 4.8 Pasangan kartu ditemukan

Setelah pemain mengklik tombol **hint**, system akan memberikan informasi kepada pemain letak pasangan kartu berada dengan tanda kartu yang berkedip-kedip. Setelah itu pemain membuka kedua kartu yang berkedip-kedip tersebut.

Jika kesempatan menggunakan tombol **hint** sudah habis, maka tombol **hint** tidak akan berfungsi dan warna tombol hint berubah menjadi abu-abu, dan pemain harus mencari sendiri pasangan kartu dengan gambar yang sama hingga semua pasangan kartu ditemukan dan permainan selesai.

Permainan selesai jika semua pasangan kartu telah ditemukan. Skor permainan dihitung dari banyaknya pemain melakukan klik, artinya semakin sedikit pemain melakukan klik maka skor semakin tinggi. Setelah permainan selesai akan muncul form untuk mengisi nama pemain yang nantinya akan dimasukkan ke dalam tabel skor. Berikut gambar tampilan form skor:



Gambar 4.9 Form Skor

Pada halaman skor pemain bisa mengisi nama pemain yang nantinya nama tersebut sekaligus skor yang didapat disimpan ke dalam tabel skor. Tabel skor dapat dilihat pada menu **Score** di halaman utama. Berikut tampilan tabel skor:

Player	Level	Clicks	Date
hehe	3	32.0	Mar 11
aji	3	41.0	Mar 17
okok	3	52.0	Mar 13
ajie	3	52.0	Mar 16

Gambar 4.10 Halaman Tabel Skor

Pemain memilih kategori skor yang ingin dilihat berdasarkan level, karena untuk tabel skor dibedakan tiap levelnya.

4.3 Uji Coba

4.3.1 Pembangkitan Permainan Menggunakan Algoritma Multiplicative CRNG

Pada uji coba pembangkitan permainan menggunakan algoritma *Multiplicative CRNG* akan dilakukan pada tiap-tiap level, yaitu *Easy*, *Medium* dan *Hard*. Penulis menunjukkan hasil *generate* bilangan acak dari variable yang terdapat pada setiap level permainan. Penulis melakukan uji coba pada setiap level permainan sebanyak tiga kali percobaan pembangkitan bilangan acak. Berikut tabel uji coba pembangkitan permainan menggunakan algoritma *Multiplicative CRNG*:

- a) Pengujian Pembangkitan Permainan Level Easy
 1. Uji Coba ke 1



Gambar 4.11 Pengujian Pembangkitan Permainan Level Easy ke 1

Tabel 4.1 Pengujian Pembangkitan Permainan Level Easy ke 1

Algoritma	Multiplicative CRNG
Parameter	$Z_0=15, a=45, m=17$
Level	Easy
Jumlah Kartu	16
Hasil Bilangan Acak	12 13 7 9 14 1 11 2 5 4 10 8 3 16 6 15

Pada gambar 4.11 menunjukkan pembangkitan permainan dengan dengan algoritma *Multiplicative CRNG* berhasil dilakukan dengan variable $z_0=15$, $a=45$, $m=17$. Hasil pengacakan dapat dilihat pada tabel 4.1 bahwa pada uji coba ke 1 untuk level *Easy* dengan jumlah kartu 16 menghasilkan bilangan acak 12 13 7 9 14 1 11 2 5 4 10 8 3 16 6 15. Tidak ada perulangan bilangan yang muncul pada pembangkitan bilangan acak.

2. Uji Coba ke 2



Gambar 4.12 Pengujian Pembangkitan Permainan Level Easy ke 2

Tabel 4.2 Pengujian Pembangkitan Permainan Level Easy ke 2

Algoritma	Multiplicative CRNG
Parameter	$Z_0=11$, $a=44$, $m=17$
Level	Easy
Jumlah Kartu	16
Hasil Bilangan Acak	8 12 1 10 15 14 4 6 9 5 16 7 2 3 13 11

Pada gambar 4.12 menunjukkan pembangkitan permainan dengan dengan algoritma *Multiplicative CRNG* berhasil dilakukan dengan variable $z_0=11$, $a=44$, $m=17$. Hasil pengacakan dapat dilihat pada tabel 4.2 bahwa pada uji coba ke 2 untuk level *Easy* dengan jumlah kartu 16 menghasilkan bilangan acak 8 12 1 10 15 14 4 6 9 5 16 7 2 3 13 11. Tidak ada perulangan bilangan yang muncul pada pembangkitan bilangan acak.

3. Uji Coba Ke 3



Gambar 4.13 Pengujian Pembangkitan Permainan Level Easy ke 3

Tabel 4.3 Pengujian Pembangkitan Permainan Level Easy ke 3

Algoritma	Multiplicative CRNG
Parameter	$Z_0=16, a=10, m=17$
Level	Easy
Jumlah Kartu	16
Hasil Bilangan Acak	7 2 3 13 11 8 12 1 10 15 14 4 6 9 5 16

Pada gambar 4.13 menunjukkan pembangkitan permainan dengan dengan algoritma *Multiplicative CRNG* berhasil dilakukan dengan variable $z_0=16$, $a=40$, $m=17$. Hasil pengacakan dapat dilihat pada tabel 4.3 bahwa pada uji coba ke 3 untuk level *Easy* dengan jumlah kartu 16 menghasilkan bilangan acak 7 2 3 13 11 8 12 1 10 15 14 4 6 9 5 16. Tidak ada perulangan bilangan yang muncul pada pembangkitan bilangan acak.

b) Pegujian Pembangkitan Permainan Level Medium

1. Uji Coba Ke 1



Gambar 4.14 Pengujian Pembangkitan Permainan Level Medium ke 1

Tabel 4.4 Pengujian Pembangkitan Permainan Level Medium ke 1

Algoritma	Multiplicative CRNG
Parameter	$Z_0=25$, $a=42$, $m=31$
Level	Medium
Jumlah Kartu	30
Hasil Bilangan Acak	27 18 12 8 26 7 15 10 17 1 11 28 29 9 6 4 13 19 23 5 24 16 21 14 30 20 3 2 22 25

Pada gambar 4.14 menunjukkan pembangkitan permainan dengan dengan algoritma *Multiplicative CRNG* berhasil dilakukan dengan variable $z_0=25$, $a=42$, $m=31$. Hasil pengacakan dapat dilihat pada tabel 4.4 bahwa pada uji coba ke 1 untuk level *Medium* dengan jumlah kartu 30 menghasilkan bilangan acak 27 18 12 8 26 7 15 10 17 1 11 28 29 9 6 4 13 19 23 5 24 16 21 14 30 20 3 2 22 25. Tidak ada perulangan bilangan yang muncul pada pembangkitan bilangan acak.

2. Uji Coba Ke 2



Gambar 4.15 Pengujian Pembangkitan Permainan Level Medium ke 2

Tabel 4.5 Pengujian Pembangkitan Permainan Level Medium ke 2

Algoritma	Multiplicative CRNG
Parameter	$Z_0=11$, $a=44$, $m=31$
Level	Medium
Jumlah Kartu	30
Hasil Bilangan Acak	19 30 18 17 4 21 25 15 9 24 2 26 28 23 20 12 1 13 14 27 10 6 16 22 7 29 5 3 8 11

Pada gambar 4.15 menunjukkan pembangkitan permainan dengan dengan algoritma *Multiplicative CRNG* berhasil dilakukan dengan variable $z_0=11$, $a=44$, $m=31$. Hasil pengacakan dapat dilihat pada tabel 4.5 bahwa pada uji coba ke 2 untuk level *Medium* dengan jumlah kartu 30 menghasilkan bilangan 19 30 18 17 4 21 25 15 9 24 2 26 28 23 20 12 1 13 14 27 10 6 16 22 7 29 5 3 8 11. Tidak ada perulangan bilangan yang muncul pada pembangkitan bilangan acak.

3. Uji Coba Ke 3



Gambar 4.16 Pengujian Pembangkitan Permainan Level Medium ke 3

Tabel 4.6 Pengujian Pembangkitan Permainan Level Medium ke 3

Algoritma	Multiplicative CRNG
Parameter	$Z_0=7, a=3, m=31$
Level	Medium
Jumlah Kartu	30
Hasil Bilangan Acak	21 1 3 9 27 19 26 16 17 20 29 25 13 8 24 10 30 28 22 4 12 5 15 14 11 2 6 18 23 7

Pada gambar 4.16 menunjukkan pembangkitan permainan dengan dengan algoritma *Multiplicative CRNG* berhasil dilakukan dengan variable $z_0=7$, $a=3$, $m=31$. Hasil pengacakan dapat dilihat pada tabel 4.6 bahwa pada uji coba ke 3 untuk level *Medium* dengan jumlah kartu 30 menghasilkan bilangan 21 1 3 9 27 19 26 16 17 20 29 25 13 8 24 10 30 28 22 4 12 5 15 14 11 2 6 18 23 7. Tidak ada perulangan bilangan yang muncul pada pembangkitan bilangan acak.

c) Pengujian Pembangkitan Permainan Level Hard

1. Uji Coba Ke 1



Gambar 4.17 Pengujian Pembangkitan Permainan Level Hard ke 1

Tabel 4.7 Pengujian Pembangkitan Permainan Level Hard ke 1

Algoritma	Multiplicative CRNG
Parameter	$Z_0=26, a=15, m=37$
Level	Hard
Jumlah Kartu	36
Hasil Bilangan Acak	20 4 23 12 32 36 22 34 29 28 13 10 2 30 6 16 18 11 17 33 14 25 5 1 15 3 8 9 24 27 35 7 31 21 19 26

Pada gambar 4.17 menunjukkan pembangkitan permainan dengan dengan algoritma *Multiplicative CRNG* berhasil dilakukan dengan variable $z_0=26$, $a=15$, $m=37$. Hasil pengacakan dapat dilihat pada tabel 4.7 bahwa pada uji coba ke 1 untuk level *Hard* dengan jumlah kartu 36 menghasilkan bilangan 20 4 23 12 32 36 22 34 29 28 13 10 2 30 6 16 18 11 17 33 14 25 5 1 15 3 8 9 24 27 35 7 31 21 19 26. Tidak ada perulangan bilangan yang muncul pada pembangkitan bilangan acak.

2. Uji Coba Ke 2



Gambar 4.18 Pengujian Pembangkitan Permainan Level Hard ke 2

Tabel 4.8 Pengujian Pembangkitan Permainan Level Hard ke 2

Algoritma	Multiplicative CRNG
Parameter	$Z_0=10, a=2, m=37$
Level	Hard
Jumlah Kartu	36
Hasil Bilangan Acak	20 3 6 12 24 11 22 7 14 28 19 1 2 4 8 16 32 27 17 34 31 25 13 26 15 30 23 9 18 36 35 33 29 21 5 10

Pada gambar 4.18 menunjukkan pembangkitan permainan dengan dengan algoritma *Multiplicative CRNG* berhasil dilakukan dengan variable $z_0=10$, $a=2$, $m=37$. Hasil pengacakan dapat dilihat pada tabel 4.8 bahwa pada uji coba ke 2 untuk level *Hard* dengan jumlah kartu 36 menghasilkan bilangan 20 3 6 12 24 11 22 7 14 28 19 1 2 4 8 16 32 27 17 34 31 25 13 26 15 30 23 9 18 36 35 33 29 21 5 10. Tidak ada perulangan bilangan yang muncul pada pembangkitan bilangan acak.

3. Uji Coba Ke 3



Gambar 4.19 Pengujian Pembangkitan Permainan Level Hard ke 3

Tabel 4.9 Pengujian Pembangkitan Permainan Level Hard ke 3

Algoritma	Multiplicative CRNG
Parameter	$Z_0=7, a=20, m=37$
Level	Hard
Jumlah Kartu	36
Hasil Bilangan Acak	29 25 19 10 15 4 6 9 32 11 35 34 14 21 13 1 20 30 8 12 18 27 22 33 31 28 5 26 2 3 23 16 24 36 17 7

Pada gambar 4.19 menunjukkan pembangkitan permainan dengan dengan algoritma *Multiplicative CRNG* berhasil dilakukan dengan variable $z_0=7$, $a=20$, $m=37$. Hasil pengacakan dapat dilihat pada tabel 4.9 bahwa pada uji coba ke 3 untuk level *Hard* dengan jumlah kartu 36 menghasilkan bilangan 29 25 19 10 15 4 6 9 32 11 35 34 14 21 13 1 20 30 8 12 18 27 22 33 31 28 5 26 2 3 23 16 24 36 17 7. Tidak ada perulangan bilangan yang muncul pada pembangkitan bilangan acak.

Hasil uji coba di atas dapat dilihat pada hasil bilangan acak, di mana semua uji coba pada tiap level mempunyai hasil yang berbeda-beda. Hasil tersebut didapat dari kombinasi pembangkitan variable a dan z_0 yang sangat variatif, sehingga bilangan acak yang didapatkan juga berbeda-beda dan hampir tidak ada kesamaan. Hasil *generate* bilangan acak tersebut kemudian diproses untuk membentuk atribut-atribut nilai pada tiap posisi indeks kartu dengan rumus seperti yang penulis jelaskan pada implementasi algoritma *Multiplicative CRNG* pada poin 4.1.1.

4.3.2 Pencarian Solusi Permainan Menggunakan Algoritma Depth First Search

Search

Penulis melakukan uji coba implementasi algoritma *Depth First Search* sebanyak tiga kali di setiap level. Berikut tabel hasil uji coba implementasi algoritma *Depth First Search* sebagai pencarian solusi / pencarian pasangan gambar:

a) Pengujian Pencarian Solusi Level Easy

1. Uji Coba ke 1



Gambar 4.20 Pengujian Pencarian Solusi Level Easy ke 1

Tabel 4.10 Pengujian Pencarian Solusi Level Easy ke 1

Algoritma	Depth First Search
Parameter	id 1=0 (indeks 0,0) id 2=301 (indeks 3,1)
Level	Easy
Jumlah Kartu	16
Waktu Pencarian	0.0000383 s

Pada gambar 4.20 menunjukkan pencarian solusi permainan dengan algoritma *Depth First Search* berhasil dilakukan dengan variable *button* id 1=0 dengan indeks acuan (0,0), yang mempunyai *attribute integer* sama dengan variable *button* id 2=301 dengan indeks tujuan (3,1). Lama proses pencarian dapat dilihat pada tabel 4.10, untuk pada uji coba ke 1 untuk level *Easy* dengan indeks acuan (0,0) dan indeks tujuan (3,1) dengan jumlah kartu 16 membutuhkan waktu 0.0000383 s.

2. Uji Coba Ke 2



Gambar 04.21 Pengujian Pencarian Solusi Level Easy ke 2

Tabel 4.11 Pengujian Pencarian Solusi Level Easy ke 2

Algoritma	Depth First Search
Parameter	id 1=100 (indeks 1,0) id 2=302 (indeks 3,2)
Level	Easy
Jumlah Kartu	16
Waktu Pencarian	0.0000383 s

Pada gambar 4.21 menunjukkan pencarian solusi permainan dengan algoritma *Depth First Search* berhasil dilakukan dengan variable *button* id 1=100 dengan indeks acuan (1,0), yang mempunyai *attribute integer* sama dengan variable *button* id 2=302 dengan indeks tujuan (3,2). Lama proses pencarian dapat dilihat pada tabel 4.11, untuk pada uji coba ke 2 untuk level *Easy* dengan indeks acuan (1,0) dan indeks tujuan (3,2) dengan jumlah kartu 16 membutuhkan waktu 0.0000383 s.

3. Uji Coba Ke 3



Gambar 4.22 Pengujian Pencarian Solusi Level Easy ke 3

Tabel 4.12 Pengujian Pencarian Solusi Level Easy ke 3

Algoritma	Depth First Search
Parameter	id 1=200 (indeks 2,0) id 2=103 (indeks 1,3)
Level	Easy
Jumlah Kartu	16
Waktu Pencarian	0.0000533 s

Pada gambar 4.22 menunjukkan pencarian solusi permainan dengan algoritma *Depth First Search* berhasil dilakukan dengan variable *button* id 1=200 dengan indeks acuan (2,0), yang mempunyai *attribute integer* sama dengan variable *button* id 2=103 dengan indeks tujuan (1,3). Lama proses pencarian dapat dilihat pada tabel 4.12, untuk pada uji coba ke 3 untuk level *Easy* dengan indeks acuan (2,0) dan indeks tujuan (1,3) dengan jumlah kartu 16 membutuhkan waktu 0.0000533 s.

b) Pengujian Pencarian Solusi Level Medium

1. Uji Coba Ke 1



Gambar 4.23 Pengujian Pencarian Solusi Level Medium ke 1

Tabel 4.13 Pengujian Pencarian Solusi Level Medium ke 1

Algoritma	Depth First Search
Parameter	id 1=0 (indeks 0,0) id 2=4 (indeks 0,4)
Level	Medium
Jumlah Kartu	30
Waktu Pencarian	0.0000817 s

Pada gambar 4.23 menunjukkan pencarian solusi permainan dengan algoritma *Depth First Search* berhasil dilakukan dengan variable *button* id 1=0 dengan indeks acuan (0,0), yang mempunyai *attribute integer* sama dengan variable *button* id 2=4 dengan indeks tujuan (0,4). Lama proses pencarian dapat dilihat pada tabel 4.13, untuk pada uji coba ke 1 untuk level *Medium* dengan indeks acuan (0,0) dan indeks tujuan (0,4) dengan jumlah kartu 30 membutuhkan waktu 0.0000817 s.

2. Uji Coba Ke 2



Gambar 4.24 Pengujian Pencarian Solusi Level Medium ke 2

Tabel 4.14 Pengujian Pencarian Solusi Level Medium ke 2

Algoritma	Depth First Search
Parameter	id 1=100 (indeks 1,0) id 2=200 (indeks 2,0)
Level	Medium
Jumlah Kartu	30
Waktu Pencarian	0.0000117 s

Pada gambar 4.24 menunjukkan pencarian solusi permainan dengan algoritma *Depth First Search* berhasil dilakukan dengan variable *button* id 1=100 dengan indeks acuan (1,0), yang mempunyai *attribute integer* sama dengan variable *button* id 2=200 dengan indeks tujuan (2,0). Lama proses pencarian dapat dilihat pada tabel 4.14, untuk pada uji coba ke 2 untuk level *Medium* dengan indeks acuan (1,0) dan indeks tujuan (2,0) dengan jumlah kartu 30 membutuhkan waktu 0.0000117 s

3. Uji Coba Ke 3



Gambar 4.25 Pengujian Pencarian Solusi Level Medium ke 3

Tabel 4.15 Pengujian Pencarian Solusi Level Medium ke 3

Algoritma	Depth First Search
Parameter	id 1=300 (indeks 3,0) id 2=204 (indeks 2,4)
Level	Medium
Jumlah Kartu	30
Waktu Pencarian	0.0000767 s

Pada gambar 4.25 menunjukkan pencarian solusi permainan dengan algoritma *Depth First Search* berhasil dilakukan dengan variable *button* id 1=300 dengan indeks acuan (3,0), yang mempunyai *attribute integer* sama dengan variable *button* id 2=204 dengan indeks tujuan (2,4). Lama proses pencarian dapat dilihat pada tabel 4.15, untuk pada uji coba ke 2 untuk level *Medium* dengan indeks acuan (3,0) dan indeks tujuan (2,4) dengan jumlah kartu 30 membutuhkan waktu 0.0000767 s.

c) Pengujian Pencarian Solusi Level Hard

1. Uji Coba Ke 1



Gambar 4.26 Pengujian Pencarian Solusi Level Hard ke 1

Tabel 4.16 Pengujian Pencarian Solusi Level Hard ke 1

Algoritma	Depth First Search
Parameter	id 1=0 (indeks 0,0) id 2=205 (indeks 2,5)
Level	Hard
Jumlah Kartu	36
Waktu Pencarian	0.0001233 s

Pada gambar 4.26 menunjukkan pencarian solusi permainan dengan algoritma *Depth First Search* berhasil dilakukan dengan variable *button* id 1=0 dengan indeks acuan (0,0), yang mempunyai *attribute integer* sama dengan variable *button* id 2=205 dengan indeks tujuan (2,5). Lama proses pencarian dapat dilihat pada tabel 4.16, untuk pada uji coba ke 1 untuk level *Hard* dengan indeks acuan (0,0) dan indeks tujuan (2,5) dengan jumlah kartu 36 membutuhkan waktu 0.0001233 s

2. Uji Coba Ke 2



Gambar 4.27 Pengujian Pencarian Solusi Level Hard ke 2

Tabel 4.17 Pengujian Pencarian Solusi Level Hard ke 2

Algoritma	Depth First Search
Parameter	id 1=100 (indeks 1,0) id 2=503 (indeks 5,3)
Level	Hard
Jumlah Kartu	36
Waktu Pencarian	0.000817 s

Pada gambar 4.27 menunjukkan pencarian solusi permainan dengan algoritma *Depth First Search* berhasil dilakukan dengan variable *button* id 1=100 dengan indeks acuan (1,0), yang mempunyai *attribute integer* sama dengan variable *button* id 2=503 dengan indeks tujuan (5,3). Lama proses pencarian dapat dilihat pada tabel 4.17, untuk pada uji coba ke 2 untuk level *Hard* dengan indeks acuan (1,0) dan indeks tujuan (5,3) dengan jumlah kartu 36 membutuhkan waktu 0.000817 s.

3. Uji Coba Ke 3



Gambar 4.28 Pengujian Pencarian Solusi Level Hard ke 3

Tabel 4.18 Pengujian Pencarian Solusi Level Hard ke 3

Algoritma	Depth First Search
Parameter	id 1=200 (indeks 2,0) id 2=2 (indeks 0,2)
Level	Hard
Jumlah Kartu	36
Waktu Pencarian	0.000417 s

Pada gambar 4.28 menunjukkan pencarian solusi permainan dengan algoritma *Depth First Search* berhasil dilakukan dengan variable *button* id 1=200 dengan indeks acuan (2,0), yang mempunyai *attribute integer* sama dengan variable *button* id 2=2 dengan indeks tujuan (0,2). Lama proses pencarian dapat dilihat pada tabel 4.18, untuk pada uji coba ke 3 untuk level *Hard* dengan indeks acuan (2,0) dan indeks tujuan (0,2) dengan jumlah kartu 36 membutuhkan waktu 0.000417 s.

Tabel hasil uji coba pencarian solusi/pencarian pasangan kartu menggunakan algoritma *Depth First Search* menunjukkan bahwa rata-rata tiap level permainan mempunyai waktu pencarian yang hampir sama. Semua pencarian dari level *Easy Medium*, sampai *Hard* membutuhkan waktu kurang dari 1 detik.

Algoritma *Depth First Search* pada permainan Pairs dibatasi dengan setiap pencarian hanya menghasilkan satu solusi. Jika solusi sudah ditemukan, maka indeks dari kartu yang sudah ketemu solusinya akan dihilangkan dari *node* matriks, sehingga pencarian akan dilanjutkan secara mangkus/mendalam. Penggunaan algoritma ini terlihat pada pencarian pasangan dari kartu sampai seluruh kartu telah diperiksa kesamaannya, kemudian baru dilanjutkan dengan kartu berikutnya.

4.3.3 Uji Coba Pengguna

Pengujian terhadap pengguna dilakukan dengan metode *Purposive Sampling*, cara penarikan sample yang dilakukan memilih subjek berdasarkan kriteria spesifik yang ditetapkan. Penulis mengambil sample dari dua jenis

sekolah yaitu sekolah yang berbasis Islam/Madrasah Ibtidaiah dan umum/Sekolah Dasar. Penulis mengambil 31 siswa Madrasah Ibtidaiah kelas 5 sedangkan untuk Sekolah Dasar penulis mengambil 18 siswa-siswi kelas 5. Penulis mengambil sampel dari 2 sekolah dengan yang berbasis Islam dan umum untuk membandingkan bagaimana tanggapan atau respon yang terjadi ketika Madrasah Ibtidaiah yang sesungguhnya sudah terdapat pelajara bahasa Arab dengan Sekolah Dasar yang tidak mendapat pelajaran bahasa Arab terhadap game Arabic Pairs. Penulis mengambil kelas 5 sebagai sample karena pada sebelumnya sudah melalui diskusi dengan kepala sekolah. Kegiatan uji coba dilaksanakan pada tanggal 23 Maret 2013 untuk Madrasah Ibtidaiah dan 15 April 2013 untuk Sekolah Dasar. Teknik uji coba dilakukan dengan cara membagi kelas menjadi beberapa kelompok yang setiap kelompoknya terdiri dari 7-8 orang siswa.

Berikut adalah hasil uji coba responden setelah memainkan game Arabic Pairs:

Tabel 4.19 Rekapitulasi Hasil Uji Coba

NO	URAIAN	YA		TIDAK	
		MI	SD	MI	SD
1	Pemain mengetahui fungsi setiap menu pada halaman utama	90,3%	100%	9,7%	0%
2	Pemain memahami petunjuk permainan pada menu bantuan	100%	100%	0%	0%
3	Pemain mengetahui cara memilih kategori	100%	100%	0%	0%
4	Pemain mengetahui cara memilih level	93,5%	88,9%	6,5%	11,1%

5	Pemain mengetahui cara memainkan permainan ini	100%	94,5%	0%	5,5%
6	Pemain mengetahui cara menggunakan bantuan menemukan pasangan kartu jika kesulitan menemukanya	93,5%	77,8%	6,5%	22,2%
7	Pemain merasa bosan memainkan <i>game</i> Arabic Pairs	3,3%	5,5%	96,7%	94,5%
8	<i>Game</i> Arabic Pairs menarik dan menyenangkan	100%	100%	0%	0%
9	Pemain lebih senang belajar menghafal mufrodad/kosakata bahasa Arab dengan <i>game</i> Arabic Pairs	100%	100%	0%	0%
10	Pemain lebih mudah belajar menghafal bentuk tulisan mufrodad/kosakata bahasa Arab dengan <i>game</i> Arabic Pairs	100%	100%	0%	0%
11	Pemain lebih mudah belajar mengucapkan mufrodad/kosakata bahasa arab dengan baik setelah memainkan <i>game</i> Arabic Pairs	100%	83,3%	0%	16,7%
12	Pemain ingin memainkan <i>game</i> Arabic Pairs lagi	100%	100%	0%	0%

Berdasarkan tabel 4.19, 31 responden Madrasah Ibtidaiah (MI) dan 18 responden dari Sekolah Dasar (SD), responden dari sebanyak 90,3% (MI) dan 100% (SD) mengetahui fungsi setiap menu pada halaman utama sedangkan

sisanya 9,7% (MI) dan 0% (SD) tidak mengetahui semua fungsi pada halaman utama. Pemahaman pemain terhadap menu bantuan mencapai nilai 100% untuk MI dan SD, artinya semua pemain mengetahui apa yang ditampilkan pada menu bantuan. Pengetahuan responden dalam memilih kategori juga mempunyai nilai 100% untuk MI dan SD, tidak ada yang menjawab tidak mengetahui dalam memilih kategori. Sebanyak 93,5% (MI) dan 88,9% (SD) responden mengetahui bagaimana memilih level permainan, sedangkan sisanya 6,5% (MI) dan 11,1% (SD) tidak mengetahui. Cara memainkan *game* ini yang cukup sederhana membuat seluruh responden MI bisa memainkan *game* ini, namun untuk SD ada 5,5% yang tidak mengetahui cara memainkan *game* ini. Adanya tombol **hint** untuk membantu pemain dalam menemukan pasangan kartu diketahui oleh 93,3% (MI) dan 77,8% (SD) responden, sedangkan 6,5% (MI) dan 22,2% (SD) tidak mengetahui fungsi tombol **hint**. *Game* Arabic Pairs yang cukup mudah dimainkan membuat sebanyak 96,7% (MI) dan 94,5% (SD) responden tidak merasa bosan memainkan *game* ini, selain itu seluruh responden menganggap bahwa *game* Arabic Pairs cukup menarik dan menyenangkan untuk dimainkan. Sebanyak 100% responden (MI dan SD) bahwa lebih senang belajar menghafal mufrodad/kosakata bahasa Arab dengan *game* Arabic Pairs. Selain itu juga lebih mudah belajar menghafal bentuk tulisan mufrodad/kosakata bahasa Arab dengan *game* Arabic Pairs. Adanya suara bahasa Arab yang keluar dari setiap kosakata dari Arabic Pairs ini juga dapat memudahkan pemain dalam belajar mengucapkan bahasa arab yang baik dan benar, hal itu dibuktikan dengan 100% reponden menjawab ya untuk MI, namun untuk SD ada 16,7% yang menjawab tidak.

Semua responden juga sangat menginginkan untuk memainkan *game* Arabic Pairs ini lebih dari satu kali.

Secara keseluruhan *game* Arabic Pairs ini sangat menarik dan menyenangkan menurut responden dengan berbagai alasan. Pembelajaran pengenalan kosakata bahasa Arab dan bagaimana pengucapan kosakata bahasa Arab juga dapat tersampaikan kepada pengguna ditunjukkan oleh 100% untuk MI. Namun ada beberapa responden dari SD yang menganggap *game* ini belum bisa untuk belajar yang efektif dalam mengenal dan mempelajari bahasa Arab, hal itu wajar karena di SD tidak ada pelajaran bahasa Arab.

4.4 Integrasi Game Arabic Pairs dengan Islam

Bahasa Arab merupakan bahasa yang digunakan dalam kitab suci Al Quran. Allah juga memerintahkan kepada umat Islam untuk mempelajarinya supaya kita termasuk golongan orang-orang yang mengetahui. Hal ini sudah dijelaskan dalam firman Allah:

كُتِبَ فُصِّلَتْ آيَاتُهُ قُرْءَانًا عَرَبِيًّا لِقَوْمٍ يَعْلَمُونَ ﴿٣﴾

Artinya:

Kitab yang dijelaskan ayat-ayatnya, yakni bacaan dalam bahasa Arab, untuk kaum yang mengetahui (Q.S Al Fushilat ayat 3)

Oleh karena itu walaupun bukan bahasa Arab bukan bahasa utama orang Indonesia, tetapi sewajarnya kita sebagai umat Islam mempelajari bahasa yang digunakan dalam agama kita. Terdapat berbagai macam cara untuk mempelajari bahasa Arab, salah satunya yaitu dengan media permainan. *Game* Arabic Pairs dirancang untuk membantu mengenalkan bahasa Arab khususnya pada anak-anak usia 7-12 tahun. *Game* yang identik dengan kesenangan saja, akan

jauh berbeda dirasakan setelah memainkan *game* Arabic Pairs, karena selain mendapat kesenangan juga sekaligus belajar mengenal bahasa Arab berikut dengan cara pengucapan yang baik dan benar.



BAB V

PENUTUP

5.1 Kesimpulan

Dari hasil implementasi dan uji coba yang telah dilakukan, maka dapat disimpulkan bahwa algoritma *Depth First Search (DFS)* dan algoritma *Multiplicative CRNG* merupakan algoritma yang cukup baik diterapkan untuk pencarian solusi permainan dan pembangkit permainan dalam *game Arabic Pairs*. Hal ini dibuktikan oleh hasil uji coba yang menunjukkan bahwa *game Arabic Pairs* mampu melakukan pencarian solusi dan membangkitkan permainan, mulai dari level *Easy, Medium* maupun *Hard*.

Game Arabic Pairs lebih tepat dimainkan untuk anak Madrasah Ibtidaiyah (MI) dan Sekolah Dasar, karena melihat hasil uji pada tabel 4.21 menunjukkan bahwa 100% dari 31 siswa-siswi MI dan 83,3% siswa-siswi SD menyatakan bahwa *game Arabic Pairs* dapat membantu dalam belajar mengenal dan mengucapkan kosakata bahasa Arab.

5.2 Saran

Penelitian ini tentu masih memiliki beberapa kekurangan, oleh karena itu untuk pengembangan penelitian ini di masa yang akan datang penulis menyarankan beberapa hal untuk bahan pengembangan, antara lain:

1. Menggunakan algoritma lain untuk pencarian solusi dan pembangkitan permainan sebagai bahan perbandingan dengan algoritma *Depth First Search* dan *Multiplicative CRNG*

2. Melakukan inovasi dari segi tampilan maupun permainan, agar pemain tidak gampang bosan untuk memainkan *game* Arabic Pairs
3. Mengembangkan *game* Arabic Pairs untuk system operasi yang lain, seperti iOS, Windows Phone , Blackberry atau yang lainnya.



DAFTAR PUSTAKA

- Atrinawati, Lovinta Happy. 2007. *Penerapan Algoritma Depth First Search (DFS) pada Pencarian Solusi Permainan "Pairs"*. Makalah Tidak Diterbitkan. Bandung: ITB Bandung
- Kiswanto, Bambang. 2009. *Efektifitas Pembelajaran Bahasa Arab Dengan Kitab Ta'limul Lughotil 'Arobiyyah Di SMP Muhammadiyah 2 Mlati, Mlati, Sleman 2008/2009*. Skripsi Tidak Diterbitkan. Surakarta: UMS Surakarta
- L, Riani. 2010. *Pembangkit Bilangan Acak, Mata Kuliah Pemodelan & Simulasi*. Bahan Kuliah Tidak Diterbitkan. Bandung: Jurusan Teknik Informatika Universitas Komputer Indonesia
- Mulyana, Eueung. 2012. *App Inventor: Ciptakan Sendiri Aplikasi Androidmu*. Yogyakarta: CV. Andi Offset
- Nofiantoro, Arix. 2011. *Analisis dan Perancangan Game "Bermain Bersama Dito & Dola"*. Skripsi Tidak Diterbitkan. Yogyakarta: AMIKOM
- Putra, Rahmatsyah. 2010. *Analisis Istilah Bahasa Arab Dalam Bank Syariah*. Skripsi Tidak Diterbitkan. Medan: Universitas Sumatera Utara
- Ali Khauli, Muhammad. 1993. *Madkhal Ilaa Ilmi Al Lughah*. Al Ardan: Darul FallahWa Tauji'
- Safaat, Nazzaruddin. 2012. *Pemrograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android*. Bandung: Informatika
- Al-Quran dan Terjemahannya. Departemen Agama RI. Bandung: Diponegoro

- Septiandri, Ali Akbar. 2011. *Implementasi Algoritma BFS dan DFS dalam Penyelesaian Token Flip Puzzle*. Makalah Tidak Diterbitkan. Bandung: ITB Bandung
- Teneng dkk. 2010. *Penerapan Algoritma Backtracking pada Permainan Math Maze*. Jurnal Informatika, VI (1): 56-67
- Yaqub, Ali Mustafa. 1990. *Nasihat Nabi kepada Pembaca dan Penghafal QUR'AN*. Jakarta: GEMA INSANI.
- Muhammad, Abubakar. 1996. *ILMU NAHWU Teori Praktis Untuk Menguasai Tata Bahasa Arab*. Surabaya: Karya Abditama
- Mustafa Al Ghalayaiani. 1980. *Jami'ud Durus Al-Arabiyah*. Bairut: Al-Maktabah Al-'Ashriyyah
- Nugroho, Adi. 2008. *Algoritma dan Struktur Data Dalam Bahasa Java*. Salatiga: CV. Andi Offset
- Nugroho, Adi. 2008. *Algoritma dan Struktur Data Dalam Bahasa C#*. Salatiga: CV. Andi Offset
- Kakiay, Thomas J. 2004. *Pengantar Sistem Simulasi*. Yogyakarta: Penerbit Andi
- Munir, Rinaldi. 2004. *Penerapan BFS dan DFS pada Pencarian Solusi*. Bahan Kuliah Tidak Diterbitkan. Bandung: ITB Bandung
- David Fox dan Roman Verhosek. 2002. *Micro Java™ Game Development*. Indianapolis: Pearson Education
- Alfthan, Erik. 2007. *Optimal strategy in the childrens game Memory*. [Online]. (www.math.kth.se, diakses tanggal 2 April 2013 pukul 15.35)
- <http://www.oup.com/oxed/children/firstwords/arabic>, diakses pada tanggal 10 Maret 2013 pukul 10.15

LAMPIRAN

Data Responden Madrasah Ibtidaiyah Islamiyah Kebonsari, Sukun, Malang

NO	NAMA	L/P	KELAS	SEKOLAH
1	Sayyidah Najwa	P	VA	MI ISLAMİYAH
2	Naufal Nizar	L	VB	MI ISLAMİYAH
3	M Hamim Jazuli	L	VB	MI ISLAMİYAH
4	Ike Fitri Widyawati	P	VB	MI ISLAMİYAH
5	Nur Elok Maharani	P	VA	MI ISLAMİYAH
6	Najib Azhar M	L	VA	MI ISLAMİYAH
7	Adinda Rahmawati	P	VB	MI ISLAMİYAH
8	Utia Rohma	P	VB	MI ISLAMİYAH
9	M Syafril S	L	VB	MI ISLAMİYAH
10	Aditya Nasrullah	L	VB	MI ISLAMİYAH
11	Azizah Nur Aini	P	VB	MI ISLAMİYAH
12	Nisa'ul Maghfiroh	P	VA	MI ISLAMİYAH
13	Fina Rahmah	P	VA	MI ISLAMİYAH
14	M Alwy	L	VB	MI ISLAMİYAH
15	Deliana Tazkya S	P	VB	MI ISLAMİYAH
16	Keren Bagus Prasetya	L	VB	MI ISLAMİYAH
17	Winandha Fajrin K	P	VB	MI ISLAMİYAH
18	Yogi Amirul	L	VB	MI ISLAMİYAH
19	Aryanni Hulalil Azkiya	P	VB	MI ISLAMİYAH
20	Kharisma Zulfa	P	VB	MI ISLAMİYAH
21	Rifqoh Nasywa Nabila	P	VB	MI ISLAMİYAH
22	Wafiq Fiqih T	P	VB	MI ISLAMİYAH
23	Rahma Zulfiyah	P	VA	MI ISLAMİYAH
24	M T Aziz	L	VB	MI ISLAMİYAH
25	Adiba Assegaf	P	VB	MI ISLAMİYAH
26	Novia Putri Arlinda	P	VB	MI ISLAMİYAH
27	Fatma Alifiatuz Zahro	P	VA	MI ISLAMİYAH
28	Nabill Dabiy	L	VB	MI ISLAMİYAH
29	M Afrizal	L	VB	MI ISLAMİYAH
30	Nur Asma'ul Khusna	P	VB	MI ISLAMİYAH
31	Sania Fauziah	P	VB	MI ISLAMİYAH

Data Responden Sekolah Dasar Negeri Banjardowo, Kabuh, Jombang

NO	NAMA	L/P	KELAS	SEKOLAH
1	Alfito Syamtabilah	L	V	SDN BANJARDOWO I
2	Endang Dwi Ratna Sari	P	V	SDN BANJARDOWO I
3	Dewi Ardiani	P	V	SDN BANJARDOWO I
4	Dewi Mayhes Pratikno	P	V	SDN BANJARDOWO I
5	Dinar Regita Putri	P	V	SDN BANJARDOWO I
6	Dony Irawan	L	V	SDN BANJARDOWO I
7	Farit Hidayat	L	V	SDN BANJARDOWO I
8	Febriyanti Endang W	P	V	SDN BANJARDOWO I
9	Jela Sastika Putri	P	V	SDN BANJARDOWO I
10	Lilik Dwi Wulandari	P	V	SDN BANJARDOWO I
11	Mei Cahyani T	P	V	SDN BANJARDOWO I
12	Oktavian Jordan F	L	V	SDN BANJARDOWO I
13	Refhisa Gehniarta	P	V	SDN BANJARDOWO I
14	Riski Mujianto	L	V	SDN BANJARDOWO I
15	Riyan Ageng Prayogo	L	V	SDN BANJARDOWO I
16	Ryo Bagus S	L	V	SDN BANJARDOWO I
17	Yayan Rizaldi	L	V	SDN BANJARDOWO I
18	Khofifah Muya Susila	P	V	SDN BANJARDOWO I

Dokumentasi Uji Coba

