

# IMPLEMENTASI ALGORITMA *LINEAR CONGRUENT METHOD* DAN ALGORITMA *SUFFIX TREE* PADA APLIKASI *CASUAL GAME* TEBAK LAGU

Peta Nurjana<sup>1</sup>, Ernawati<sup>2</sup>, Aan Erlansari<sup>3</sup>

<sup>1,2,3</sup>Program Studi Teknik Informatika, Fakultas Teknik, Universitas Bengkulu  
Jl. WR. Supratman Kandang Limun Bengkulu 38371A INDONESIA  
(telp: 0736-341022; fax: 0736-341022)

<sup>1</sup>petanurjana@gmail.com,  
<sup>2</sup>ernawati@unib.ac.id,  
<sup>3</sup>aan\_erlanshari@unib.ac.id

*Abstrak* : Penelitian ini membangun aplikasi *casual game* tebak lagu berbasis android sebagai bentuk pemanfaatan teknologi untuk sarana hiburan, budaya dan pengetahuan lagu Indonesia. Kombinasi algoritma pengacakan *Linear Congruent Method* dan algoritma *Suffix Tree* dalam mengupayakan peningkatan keunggulan *game* tebak lagu. Algoritma *Linear Congruent Method* digunakan pada pengacakan lagu untuk mendapatkan periode lagu acak yang maksimal pada *game* tebak lagu dan Algoritma *Suffix Tree* digunakan untuk pencarian string dalam menebak lirik lagu pada *game* tebak lagu. Hasil akhir dari aplikasi ini adalah kombinasi algoritma pengacakan *Linear Congruent Method* dan algoritma *Suffix Tree* dalam mengupayakan peningkatan keunggulan *game* tebak lagu dalam menampilkan lagu acak yang lebih bervariasi. Peneliti melakukan pengujian *white box* dengan membuat perkiraan logika yang kompleks dan pengujian *black box* dengan teknik pengujian yang membagi domain *input*, menentukan kasus pengujian dengan mengungkapkannya kelas-kelas kesalahan.

Kata kunci : *Game* tebak lagu, android, *Linear Congruent Method*, *Suffix Tree*.

*Abstract* : This study builds applications *casual game* guess the song based on Android as a form of the use of technology for entertainment, culture and knowledge of Indonesian songs. The combination of *Linear Congruent Method* and *Suffix Tree* can improve the song guessing game advantage. Algorithms *Linear Congruent Method* randomization used in the song to get the maximum period a random song on the song guessing game and *Suffix Tree* algorithm used for a search string in a guessing game guess the lyrics in the song. The end result of this application is a combination of *Linear Congruent Method* of randomization algorithms and algorithms *Suffix Tree* in the effort to improve excellence in the game guess the song displays a random song is more varied. Researchers do *white-box* testing to make an estimate of complex logic and *black box* testing technique that divides the input domain, specify test cases by revealing error classes.

**Keywords:** *guessing game* song, android, *Linear Congruent Method*, *Suffix Tree*.

## I. PENDAHULUAN

Musik adalah seni pengungkapan gagasan melalui bunyi yang unsur dasarnya berupa melodi, irama, dan harmoni dengan unsur pendukung berupa bentuk gagasan, sifat dan warna bunyi. Musik dan lagu adalah hal yang tidak terpisahkan, melalui musik dan lagu dapat menggambarkan karakter budaya dari suatu bangsa khususnya Indonesia. [1]

Suatu algoritma adalah urutan instruksi yang tidak ambigu untuk menyelesaikan suatu masalah, yaitu untuk mendapatkan hasil yang disyaratkan untuk setiap *input* logis dalam suatu jangka waktu tertentu. Algoritma adalah urutan langkah-langkah logis penyelesaian masalah yang disusun secara sistematis. [2] Di antara algoritma tersebut adalah algoritma pengacakan LCM (*Linear Congruent Method*) dan algoritma *suffix tree*.

LCM (*Linear Congruent Method*) adalah proses menurunkan secara acak nilai variabel tidak pasti secara berulang-ulang untuk mensimulasikan model. Metode LCM bertujuan untuk memunculkan variabel acak melalui sampling dari distribusi probabilitas. Dokumen representasi adalah masalah utama dalam pencarian informasi *frasa* berdasarkan dokumen dan algoritma pengambilan yang menggabungkan keuntungan dari dua algoritma, yaitu *suffix tree* dan Algoritma *Boyer-Moore* dengan hasil yang menyatakan bahwa algoritma *suffix tree* sebagai struktur data yang efisien untuk menyimpan teks besar dan lebih cepat, tetapi algoritma *Boyer-Moore*, lebih cepat dalam pencarian. [3].

Berdasarkan uraian dan permasalahan diatas penulis bermaksud untuk mengimplementasikan Algoritma *Linear Congruent Method* dan Algoritma *Suffix Tree* pada Aplikasi *Casual Game*

Tebak Lagu. Implementasian Algoritma *Linear Congruent Method* dan Algoritma *Suffix Tree*, pada aplikasi *game* tebak lagu sengaja dipilih karena aplikasi *game* tebak lagu memiliki objek yang sesuai untuk algoritma tersebut. Algoritma *Linear Congruent Method* digunakan pada pengacakan lagu yang akan diputar pada *game* tebak lagu, yaitu pada *menu* tebak judul lagu. Algoritma *Suffix Tree* digunakan untuk menebak lirik lagu pada *game* tebak lagu. Algoritma *suffix tree* ini akan membangun pencocokan pada teks lirik lagu untuk menu tebak lirik, lirik yang harus ditebak merupakan potongan lirik yang kosong dari satu judul lagu

## II. LANDASAN TEORI

### A. Algoritma *Linear Congruent Method*

Algoritma LCM (*Linear Congruent Method*) adalah proses menurunkan secara acak nilai variabel tidak pasti secara berulang-ulang untuk mensimulasikan model. Metode LCM bertujuan untuk memunculkan variabel acak melalui sampling dari distribusi probabilitas. LCM didefinisikan dalam persamaan sebagai berikut:

$$X_n = (a \times X_{n-1} + b) \text{ mod } m$$

$n$  = bilangan asli

$X_n$  = bilangan acak ke- $n$  dari deretnya

$X_{n-1}$  = bilangan acak sebelumnya

$a$  = factor pengali

$b$  = penambah (*increment*)

$m$  = modulus

( $a, b, \text{ dan } m$  semuanya *konstans*)

Kunci pembangkit adalah  $X_0$  yang merupakan nilai pertama kali yang muncul pada pengacakan yang di dapat secara acak. LCM mempunyai periode tidak lebih besar dari  $m$ , dan pada kebanyakan kasus periodenya kurang dari itu.

LCM mempunyai periode penuh  $(m - 1)$  jika memenuhi syarat berikut:

1.  $b$  relative prima terhadap  $m$ .
2.  $a - 1$  dapat dibagi dengan semua faktor prima dari  $m$
3.  $a - 1$  adalah kelipatan 4 jika  $m$  adalah kelipatan 4
4.  $m > maks(a, b, X_0)$
5.  $a > 0, b > 0$

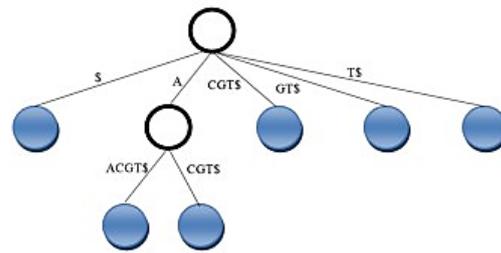
Meskipun LCM secara teoritis mampu menghasilkan bilangan acak yang lumayan, namun sangat sensitif terhadap pemilihan nilai-nilai  $a$ ,  $b$ , dan  $m$ . pemilihan nilai-nilai yang tidak sesuai dapat mempengaruhi implementasi pada LCM. [4]

#### B. Algoritma *Suffix Tree*

*Suffix tree* adalah struktur data yang merepresentasikan *suffix* (akhiran) dari string sedemikian sehingga memudahkan implementasi tertentu dengan cepat untuk berbagai operasi pada string. *Suffix tree* dari string  $S$  adalah pohon yang sisinya (*edge*) diberikan label *string* yang merupakan *suffix* dari  $S$  dan sesuai dengan tepat satu jalur dari *root* ke *leaf*. *Suffix tree* untuk string  $S$  yang panjangnya  $n$  didefinisikan sebagai suatu pohon sedemikian sehingga:

- (a) Lintasan dari *root* ke *leaf* memiliki hubungan satu-satu dengan *suffix* dari  $S$ .
- (b) Setiap *edge* (sisi) diisi string tak kosong.
- (c) Semua simpul dalam (*internal node*) kecuali *root* memiliki paling sedikit dua anak (*child*).

Di dalam *suffix tree*, string  $S$  selalu ditambahkan dengan sebuah terminal symbol yang tidak termasuk di dalam substring  $S$ , biasanya dilambangkan dengan \$.



Gambar 2.1 *Suffix Tree* untuk string 'AACGT'

*Suffix tree* juga menyediakan salah satu solusi waktu *linear* pertama untuk masalah longest common substring. Ini menyebabkan *suffix tree* sebuah string biasanya membutuhkan ruang yang jauh lebih banyak dari pada menyimpan string itu sendiri. [5]

### III. METODE PENELITIAN

#### A. Jenis Penelitian

Jenis penelitian yang digunakan adalah penelitian terapan. Penelitian terapan bertujuan untuk memperoleh penemuan-penemuan yang berkenaan dengan aplikasi/penerapan teori-teori tertentu. Penelitian yang akan dilakukan yaitu penelitian tentang Implementasi Algoritma *Linear Congruent Method* dan Algoritma *Suffix Tree* pada Aplikasi *Casual Game* Tebak Lagu. Algoritma *Linear Congruent Method* digunakan pada pengacakan lagu yang akan diputar pada *game* tebak lagu, yaitu pada menu tebak judul lagu dan algoritma *Suffix Tree* digunakan untuk menebak lirik lagu pada menu tebak lirik lagu.

#### B. Metode Pengumpulan Data

Metode pengumpulan data yang dilakukan untuk mendapatkan data serta informasi untuk mendukung penyempurnaan hasil dari penelitian ini antara lain:

#### 1. Studi Pustaka

Penelitian ini dilakukan dengan mencari informasi melalui internet serta mempelajari jurnal yang berhubungan dengan *game*, musik dan pemograman Android serta mempelajari perangkat lain yang diperlukan dalam perancangan tugas akhir ini. dari bahan bahan tersebut diambil teori - teori yang dapat dijadikan landasan untuk menganalisa masalah yang diketemukan dalam penelitian.

#### 2. Studi Literatur

Pengumpulan data dengan cara mengumpulkan literatur, jurnal, paper, tesis dan bacaan – bacaan yang ada kaitannya dengan judul penelitian..

### C. Metode Pengembangan Sistem

Pengembangan aplikasi *game* tebak lagu yang digunakan dalam penelitian ini adalah model *prototyping*. Berikut tahap – tahap yang dilakukan didalam membangun aplikasi *Casual Game* Tebak Lagu menggunakan model *prototyping*:

#### 1. Analisa Kebutuhan

##### a. Perencanaan

Dimulai dengan pengumpulan data dan informasi dari aplikasi yang akan dibangun. Data masukan yang dibutuhkan dalam aplikasi ini adalah informasi, data seputar lagu dan lirik.

##### b. Kebutuhan data keluaran

Adapaun data keluaran berupa audio atau lagu sesuai data masukan dan yang dibangun dalam *game* tebak lagu.

##### c. Kebutuhan antarmuka

Kebutuhan antarmuka pada aplikasi adalah tampilan dengan kemudahan pengguna saat mengakses aplikasi *game* tebak lagu.

#### 2. Perancangan Sistem dan Aplikasi

Tahap ini bertujuan memperkirakan pengkodean sistem dan merancang tampilan dari sistem. Tahap ini juga membantu dalam mengspesifikasikan kebutuhan perangkat keras dan sistem.

#### 3. Penerapan dan Pengujian Sistem

Tahap penerapan dan pengujian program adalah hasil dari tahapan implementasi. Hasil pemrograman di tahap sebelumnya diterapkan pada tahap ini dan diuji kelayakannya. Proses pengujian yang dilakukan pada sistem yang menggunakan dua metode pengujian, yaitu *white box testing* dan *black box testing*.

#### 4. Penggunaan dan Pemeliharaan

Setelah aplikasi selesai maka pengguna akan menggunakan aplikasi. Jika terdapat pengembangan fungsional dari aplikasi yang diinginkan oleh pengguna, maka akan dilakukannya pemeliharaan.

### IV. ANALISIS DAN PERANCANGAN

#### A. Identifikasi Masalah

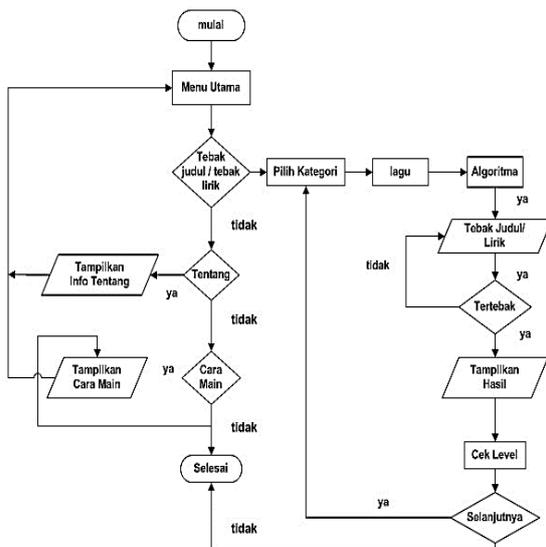
Saat ini permainan tebak lagu telah banyak beredar dan dengan mudahnya dapat diunduh di *playstore*, *appstore*, ataupun situs pemasaran aplikasi lainnya. Namun, *game* tebak lagu yang mengangkat lagu – lagu wajib nasional dan lagu – lagu asli Indonesia sangat jarang ditemui. Masalah yang dihadapi dalam pembuatan aplikasi *game* tebak lagu adalah bagaimana permainan ini dapat menghibur dan dapat dimainkan bukan hanya untuk hiburan semata, melainkan juga untuk memberikan nilai positif kepada para pemain. Tetapi juga dapat menambah wawasan terhadap lagu lagu Indonesia hingga lagu wajib nasional yang disajikan di dalam *game* tebak lagu.

## B. Analisis Sistem

Analisis sistem adalah bagian dari penelitian yang menganalisis sistem yang ada, dimana fungsinya adalah untuk merancang sistem baru atau memperbaharui sistem yang sudah ada. Pada bagian ini merupakan bagian terpenting karena hasil sistem yang akan dibangun tergantung pada analisis yang dilakukan.

### 1. Alur Sistem

Secara garis besar cara kerja sistem yang dibangun ditampilkan dalam Gambar 4.1. Alur Aplikasi *Casual Game* Tebak Lagu dengan menerapkan algoritma *Linear Congruent Method* dan algoritma *Suffix Tree* ini menyajikan permainan tebak lagu dan tebak lirik dengan beberapa kategori *genre* lagu yaitu lagu pop, dangdut dan lagu wajib nasional.



Gambar 4.1 Diagram Alur Sistem Implementasi Algoritma *Linear Congruent Method* dan Algoritma *Suffix Tree*

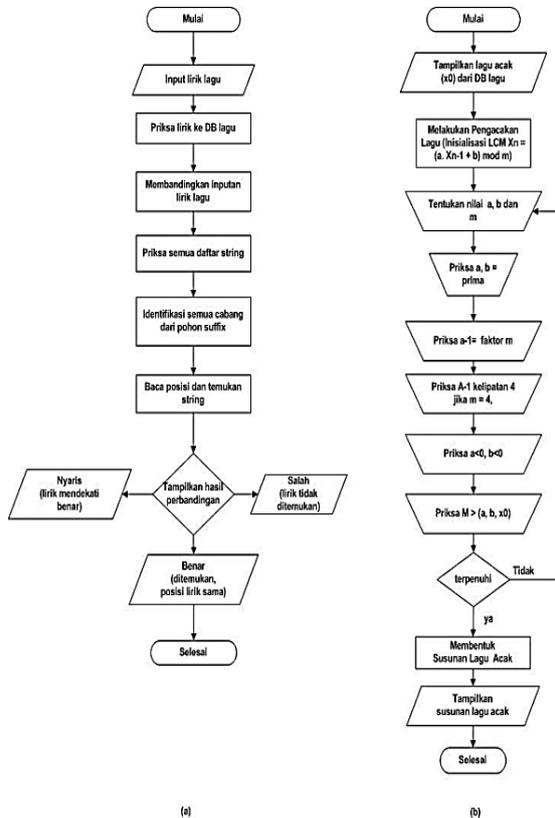
Pada gambar 4.1 diawali dengan Mulai yang menggunakan simbol terminator yang menggambarkan kegiatan awal program. Setelah itu, sistem akan menampilkan *Menu Utama*. Setelah tampil *Menu Utama*, pengguna disediakan beberapa pilihan menu yang dapat dipilih. Setiap menu yang ada diwakili oleh

simbol decision yang berfungsi menggambarkan suatu keputusan atau tindakan yang harus diambil pada kondisi tertentu (Ya/Tidak). Jika pengguna memilih menu Mulai, maka selanjutnya adalah memilih kategori genre lagu. Ketika pengguna memilih genre lagu maka lagu akan diputar atau ditebak dan sistem akan memproses algoritma untuk mengacak atau mencocokkan prasa pada lagu atau lirik dan kemudian sistem menampilkan hasil dari penggunaan algoritma tersebut. Jika pengguna berhasil melewati tantangan tersebut maka sistem akan menampilkan hasil berupa informasi, namun jika belum berhasil pengguna tetap akan kembali mengulang menebak lagu yang ditunjukkan dengan simbol *line connector* yang mengarah ke “Tebak Lagu”. Ketika pengguna berhasil maka level selanjutnya dapat dimainkan. Berikutnya terdapat decision “Selanjutnya” untuk meneruskan permainan dengan memilih kategori kembali atau melanjutkan level yang tersedia, atau tidak melanjutkannya dan permainan selesai. Jika pengguna tidak memilih Menu Utama tapi memilih menu Tentang maka sistem akan menampilkan info tentang pembuat aplikasi. Jika pengguna tidak memilih menu Tentang tetapi memilih menu Cara Main maka sistem akan menampilkan info tentang Cara main atau menggunakan aplikasi. Jika tidak memilih Menu cara main juga maka alur sistem selesai yang disimbolkan dengan terminator “Selesai”.

### 2. Alur Algoritma *Linear Congruent Method* dan Algoritma *Suffix Tree*

Pada sistem yang dibangun ini terdapat dua algoritma yang digunakan yaitu algoritma Algoritma *Linear Congruent Method* ( LCM) dan *Suffix Tree*. Untuk melihat langkah –

langkah kedua algoritma *flowchart* diagram dari algoritma *Linear Congruent Method* dan Algoritma *Suffix Tree* digambarkan pada Gambar 4.2.



Gambar 4.2 Diagram Alur Implementasi Algoritma *Suffix Tree* (a) dan Algoritma *Linear Congruent Method* (b)

Pada gambar 4.2, algoritma dimulai dengan menentukan pilihan *menu* yaitu tebak judul atau tebak lirik lagu. Pada tebak lirik lagu algoritma *Suffix Tree* (a) akan melakukan pencarian lirik untuk mencocokkan lirik yang di *input* kan oleh user. Algoritma *Suffix Tree* ini membaca string lirik lagu yang ada di database untuk menampilkan hasil lirik dalam pencarian dan pencocokkan lirik lagu yang di *input* kan.

Pada tebak judul lagu algoritma yang digunakan adalah algoritma *Linear Congruent Method* (LCM) yang akan mengacak lagu

dengan inisialisasi nilai yang telah ditentukan sebelumnya dengan rumus  $X_n = (a.X_{n-1} + b) \text{ mod } m$ . LCM ini akan mengacak lagu yang akan ditebak oleh *user* sehingga lagu yang akan muncul setiap permulaan permainan pada setiap *level* akan berbeda..

## V. PEMBAHASAN

Setelah proses analisis dan perancangan sistem selesai dilakukan, maka langkah selanjutnya adalah proses pengujian terhadap sistem yang telah dibangun. Pengujian dilakukan untuk mengevaluasi kesalahan-kesalahan dan memastikan apakah sistem yang dibangun telah berjalan dengan baik. Berikut akan dibahas tentang pengujian dari Implementasi Algoritma *Linear Congruent Method* dan Algoritma *Suffix Tree* pada Aplikasi Casual *Game* Tebak Lagu

### A. Implementasi Antar Muka

#### 1. Menu *Home*

Menu utama aplikasi merupakan *menu* yang akan tampil setelah *splash screen* dijalankan yang ditunjukkan pada Gambar 5.3.



Gambar 5. 1 Tampilan *Menu* Utama aplikasi Tebak Lagu

Pada gambar 5.1 merupakan tampilan *menu* utama yang menampilkan empat pilihan *menu* yang dapat pengguna pilih, yaitu *menu* tebak lirik dan tebak lagu yang merupakan *menu*

yang akan menampilkan *game* tebak lagu. *Menu* cara main pada aplikasi merupakan *menu* yang memberikan informasi bagaimana cara menggunakan aplikasi dan *menu* tentang merupakan *menu* yang menjelaskan tentang aplikasi *game* tebak lagu.

#### B. Halaman *Menu* Tebak Judul

Pada halaman *menu* tebak judul terdapat dua pilihan menu yang dapat dipilih yaitu menu kategori lagu pop dan kategori lagu dangdut. Masing – masing menu memiliki jumlah lagu dan *level* yang sama hanya dibedakan pada lagu yang disediakan. Total jumlah lagu yang digunakan pada keduanya adalah 120 lagu. Berikut ini merupakan gambar 5.2 *menu* tebak judul.



Gambar 5. 2 Tampilan Game Tebak Judul

Pada Gambar 5.2 pengguna dapat memilih pilihan kategori lagu yang diinginkan. Kategori lagu pop dan kategori lagu dangdut masing – masing memiliki 3 *level* pada masing – masing *level* berisi jumlah lagu yang berbeda. Pengguna harus menyelesaikan *level* pertama untuk melanjutkan ke *level* berikutnya.

#### C. Halaman *Menu* Tebak Lirik

Pada halaman *menu* tebak Lirik terdapat 3 *level* lagu wajib nasional, masing – masing *level*

memiliki jumlah lagu yang berbeda pada *level* pertama memiliki 4 lagu, *level* kedua 7 lagu, dan *level* ketiga 9 lagu yang dapat ditebak liriknya. Jumlah data yang digunakan pada *game* tebak lirik ini adalah 40 data berupa data lirik lagu dan instrument musik lmasing – masing lagu. Berikut ini merupakan gambar 5.3 yang merupakan tampilan pada tebak lirik.



Gambar 5. 3 Tampilan *game* Tebak Lirik

Pada gambar 5.3 pengguna harus menekan tombol mulai untuk mendengarkan instrumen lagu wajib nasional. Pengguna harus mengetik lirik yang tepat untuk bagian yang kosong untuk satu lirik dan menekan tombol cek untuk mengetahui benar atau salah lirik yang diketik tersebut. Pengguna memiliki tiga kesempatan untuk salah dalam menebak lirik dalam satu *level*.

#### D. Implementasi *Algoritma Linier Congruen Method*

Dalam permainan tebak lagu yang dirancang ada beberapa *level* yang tersedia. Masing – masing *level* memiliki jumlah lagu yang berbeda,

level 1 memiliki 10 buah lagu, level 2 memiliki 20 buah lagu dan level 3 memiliki 30 buah lagu dalam satu kategori genre lagu.

Penerapan Algoritma *Linier Congruen Method* Level 1. Lagu yang dimunculkan pada permainan ini ditampilkan secara acak menggunakan metode *linier congruent method*. Berikut ini pembahasan bagaimana LCM menampilkan kesepuluh pertanyaan pada level 1. LCM mempunyai periode penuh ( $m - 1$ ) jika memenuhi syarat berikut.

1.  $b$  relative prima terhadap  $m$
2.  $a - 1$  dapat dibagi dengan semua faktor prima dari  $m$
3.  $a - 1$  adalah kelipatan 4 jika  $m$  adalah kelipatan 4.
4.  $m > maks(a, b, X_0)$
5.  $a > 0, b > 0$

Sesuai dari syarat diatas untuk mendapatkan periode penuh dari LCM, tentukan:

$$a = 2,$$

$$b = 5$$

$$m = 11$$

$$X_0 = 4 \text{ (diambil melalui proses acak)}$$

Periksa pemenuhan sarat:

1.  $b$  relative prima terhadap  $m$   
 $b = 5$  relative prima terhadap  $m = 11$
2.  $a - 1$  dapat dibagi dengan semua faktor prima dari  $m$   
 $2 - 1 = 1$  dapat dibagi dengan semua faktor prima dari 11
3.  $a - 1$  adalah kelipatan 4 jika  $m$  adalah kelipatan 4  
 $m = 11$  bukan kelipatan 4
4.  $m > maks(a, b, X_0)$   
 $11 > maks(2, 5, 4)$
5.  $a > 0, b > 0$

$$2 > 0, 5 > 0$$

Syarat terpenuhi maka langkah selanjutnya adalah menghitung proses pengacakan sesuai rumus pada Persamaan 2.1:

$$X_n = (a \cdot X_{n-1} + b) \text{ mod } m$$

$$X_0 = 4 \text{ (nilai 4 diambil melalui proses acak)}$$

Tabel 5. 1 Hasil perhitungan LCM level 1

No	$X_n = (a \cdot X_{n-1} + b) \text{ mod } m$	Hasil Random
1	$X_1 = (2 \cdot X_0 + 5) \text{ mod } 11$ $= (2 \cdot 4 + 5) \text{ mod } 11$	2
2	$X_2 = (2 \cdot X_1 + 5) \text{ mod } 11$ $= (2 \cdot 2 + 5) \text{ mod } 11$	9
3	$X_3 = (2 \cdot X_2 + 5) \text{ mod } 11$ $= (2 \cdot 9 + 5) \text{ mod } 11$	1
4	$X_4 = (2 \cdot X_3 + 5) \text{ mod } 11$ $= (2 \cdot 1 + 5) \text{ mod } 11$	7
5	$X_5 = (2 \cdot X_4 + 5) \text{ mod } 11$ $= (2 \cdot 7 + 5) \text{ mod } 11$	8
6	$X_6 = (2 \cdot X_5 + 5) \text{ mod } 11$ $= (2 \cdot 8 + 5) \text{ mod } 11$	10
7	$X_7 = (2 \cdot X_6 + 5) \text{ mod } 11$ $= (2 \cdot 10 + 5) \text{ mod } 11$	3
8	$X_8 = (2 \cdot X_7 + 5) \text{ mod } 11$ $= (2 \cdot 3 + 5) \text{ mod } 11$	0
9	$X_9 = (2 \cdot X_8 + 5) \text{ mod } 11$ $= (2 \cdot 0 + 5) \text{ mod } 11$	5
10	$X_{10} = (2 \cdot X_9 + 5) \text{ mod } 11$ $= (2 \cdot 5 + 5) \text{ mod } 11$	4
11	$X_{11} = (2 \cdot X_{10} + 5) \text{ mod } 11$ $= (2 \cdot 4 + 5) \text{ mod } 11$	2

Dari Tabel 5.1 sesuai dengan  $a = 2, b = 5$ , dan  $m = 11$  dan  $X_0 = 4$  diperoleh  $X_1 = 2$  dan seterusnya sehingga didapat bilangan acak sampai 10 periode. Bilangan acak tersebut didalam aplikasi *game* akan dijadikan acuan untuk



menggunakan metode pengacakan lainnya seperti *Fisher Yates Shuffle* dan algoritma *Boyer Moore*.

2. Sistem ini dapat dilakukan pengembangan dengan pembaruan lagu dan *menu* sehingga dapat membuat *game* yang lebih kompleks.

#### REFERENSI

- [1] Muhammad Syafiq. 2005. *Ensiklopedia Musik Klasik*, Yogyakarta: Adicitra Karya Nusa.
- [2] Rinaldi Munir. 2002, *Algoritma dan Pemrograman dalam Bahasa Pascal dan C*, Informatika, Bandung
- [3] B.Ganga, 2014. *Phrase Based Document Retrieving by Combining Suffix Tree index data structure and Boyer-Moore faster string Searching algorithm. International Journal of Advancements in Research & Technology*
- [4] Afrian 2014. *Perancangan Aplikasi Game Asah Otak Tebak Kata Berbasis Android Dengan Menggunakan Metode Linear Congruent Method (LCM)*. Pelita Informatika Budi Darma, Volume VI No 1.
- [5] Rahim Aditrian. 2013, *Penyusunan Overlap Graph Menggunakan Suffix Tree Pada DNA Sequence*, Institut Pertanian Bogor.