Use of Algorithm Linear Congruential Generator (LCG) on the TOEFL Test Application

Mahdi
Lhokseumawe State Polytechnic, Information Technology Department, Aceh-Indonesia.

Corresponding author: mahdi@pnl.ac.id

Abstract. English as one of the languages most used by the world's population to communicate. The English language proficiency testing model for one of them is the TOEFL (Test of English as Foreign Language). TOEFL is intended as a measuring tool or evaluation of a person's ability to understand English. The use of internet technology advances in developing TOEFL learning media can provide other alternatives in studying the TOEFL. The existence of this website is expected to provide various facilities for the public, especially internet users in studying the TOEFL, so that the TOEFL is no longer an obstacle and can encourage someone's interest in learning English. The web-based TOEFL application system can provide all types of information needed as well as the process of listing examinees, recapitulating participant data and printing certificates in a structured manner. The method used in this application to scramble the problem is the Linear Congruential Generator (LCG) method which can produce a series of pseudo random numbers based on several adjusted repetitions.

1. Introduction
The development of information technology is very rapid, online-based applications have sprung up as well. One of them is the increase in information technology due to the presence of the internet. With the internet we can receive and access information in various formats from all corners of the world. The presence of the internet can also provide convenience in the world of education, this is seen by the many sites that provide online learning and exam media that are increasingly interactive and easy to learn. Online sites can not only be used as information media, but also various learning systems can be done on the internet, such as TOEFL learning.

The TOEFL (Test of English and Foreign Language) is a benchmark for a person's ability to speak English. TOEFL is used for those who are preparing to study abroad, but at this time the TOEFL is also used by educational institutions or large companies to find out how much of a person's English language abilities are either students or employees. In the TOEFL itself, there are 3 types of tests that are tested, namely Listening is intended to measure the ability to hear and listen to English, Structure to measure English language skills with book grammar and Reading to measure reading comprehension skills. The method used in this application to scramble the problem is the Linear Congruential Generator (LCG) method which can produce a series of pseudo random numbers based on several repetitions that have been adjusted.
1.1. TOEFL (Test of English as a Foreign Language).
TOEFL (Test of English as a Foreign Language) is a standardized test that is usually used to determine the English language skills of a person as a speaker of a foreign language. This test was made by an institution called Educational Testing Service or more popular with the abbreviation ETS [1]. In another source, Brown also argued that the TOEFL was one of the commercially produced English proficiency tests. Like Brown, Sharpe also affirms that the TOEFL is a test of English as a foreign language. Until now, TOEFL is one of the most widely used forms of English language tests. This shows that the TOEFL is a reliable test. Aside from its function and role as an English proficiency test and its commercial nature, in general the TOEFL is included in the well-standardized test category. In the TOEFL there are several parts of the test that are the basis, namely:

1. Listening Comprehension. The first part of the TOEFL is the Listening Comprehension Section, where test participants listen to the conversation, participants are asked to listen to the questions uttered only once through audio media where the questions are not written on the question sheet or booklet, then participants answer questions based on what has been listened to. In the Listening Comprehension section this section consists of 50 questions for a period of 30-40 minutes. The Listening Comprehension Section consists of 3 parts, namely:
   a. Part A. Consisting of 20 questions (1-20), in this session, participants are asked to choose the sentence that best matches the meaning heard from the audio.
   b. Part B. It consists of 15 questions (21-35), in this session there is a short conversation between two people and each of them is followed by a question spoken by a third person. Participants are asked to choose the answer that best matches the conversation.
   c. Part C. Consists of 15 questions (36-50) whose contents are quite long conversations, and each is followed by several questions. We are asked to choose the most appropriate and appropriate answer to the conversation.

2. Structure and Written Expression. The second part of the TOEFL is Structure and Written Expression. This section consists of 50 questions about formal English or languages that use grammar rules. The Structure and Written Expression section are divided into 2 parts, Incomplete Sentence which consists of 25 questions and Error Recognition which consists of 25 questions. The time provided in answering questions in this section is 25 minutes to complete the sentence and determine the errors that exist.

3. Reading Comprehension. The last part of the TOEFL is Reading Comprehension, in this section it consists of 50 questions with a time period of 45 minutes. In this session there are some short readings, each of which is followed by a question, where each reading is not the same length and is also followed by questions that are not the same number. The topics discussed in this section vary, but basically are almost always repeated.

1.2. TOEFL score conversion.
In the TOEFL assessment, the conversion table is used to convert the values in each session of the question then calculated by the formula below:
1. Add the number of conversion results Total = listening scores + writing score + reading score.
2. Divide the sum above with 3 Results = Amount: 3.
3. Multiply the Distribution Results with 10 TOEFL Scores = Distribution Results x 10.

1.3 Algorithm Random Number Generator (RNG).
Random numbers are a number that results from a process, whose output cannot be predicted, and the same number cannot be generated sequentially. The process of generating a random number is called a random number generator. Although it seems quite simple, by definition, but in reality, it is quite difficult to produce truly random numbers [2]. Today, after the emergence of computational random number generators, more and more, lotore games, use RNGs instead of more traditional drawing methods, such as using ping-pong or rubber balls. RNGs are also used today to determine the possibility of modern slot machines. Some computational methods for random number generator exist,
but often fall short of the purpose of true randomness - even though they may meet, with various successes, some statistical tests for randomness are intended to measure how their unexpected results are (i.e. For their clear degree) Randomness testing aims to determine whether a number of numbers are generated by a random generator or not.

There are two main approaches to generating random numbers using computers: True Random Number Generators (TRNGs) and Pseudo-Random Number Generators (PRNGs). True Random Number Generators (TRNGs) are extracts of randomness from physical phenomena and introduce them into computers.

1.4. Linear Congruential Generator (LCG)
Linear Congruential Generators (LCG) are algorithms that are quite popular in various types of applications. An LCG is capable of producing a series of pseudo random numbers based on several adjusted repetitions [3]. The simple form of the LCG algorithm is as follows:
\[X_{n+1} = (a \cdot X_n + b) \mod m, \quad n \geq 0;\]

A constant in the above algorithm is called a multiplier, the constant \(b\) is called an increment, and the constant \(m\) is called modulus. The resulting series of numbers is: \(X_0, X_1, X_2, \ldots, X_m\) where \(X_0\) is the generator key or also called bait (seed). The \(a\), \(b\), \(m\), and \(X_0\) constants are parameters of the LCG algorithm. The maximum length of the pseudo random number of the LCG algorithm is as much as \(m\). Even in most cases the period produced is less than \(m\).

2. Research methods
This research requires various kinds of reference data and input for further analysis. The data is the basic theoretical data needed in research preparation. Observations made before and during the research process take place, the data needed in this case are about TOEFL exam questions. These data can be obtained by conducting surveys in the field and through TOEFL books. The existence of supporting data for this study will make it easier for researchers to understand how to build a new system based on theories that have been studied before.

2.1 System Design.
Designing is the second stage that is carried out after analysis in building a system. System design is done after getting a clear picture of what needs to be done. The purpose of system design is to meet the needs of users (users) regarding a clear picture of the system design that will be created and implemented. The system design method carried out in this project consists of several stages, namely: making a Context Diagram, Entity Relationship Diagram (ERD), designing Data Flow Diagrams (DFD), designing a table structure, flowchart and designing the user interface as the interface used for displays of the TOEFL test application system.

2.1.1. Context Diagram. The outline of the organization of this application system that will be built and shows the relationship between entities that are directly involved with the system and also is a depiction of the data streams that exist on the system that will be made as a whole with the aim of identifying the system. Context diagrams also show this application system as a whole. The following is the context diagram of the TOEFL test application system in Figure 1.

2.1.2 Entity Relationship Diagram (ERD). Entity Relationship Diagram (ERD) serves to describe the relationship between one entity and another entity, while the relationship between entities contained in the TOEFL test application system is intended in Figure 2.
2.1.3. Database Design

1. **Table of Users.** The users table is used to store all data of users who have logged into the system. The users table has a User_id, user_name, name, email, password, remember_token and hak_akses.

2. **Table Schedule.** The schedule table is used to store the types of tests that will be carried out by TOEFL test participants at registration. The schedule table has an id_jadwal, jenis_test, date_test and status.

3. **Score table.** The score table is used to store all data values of participants who have logged into the system and have entered the test session. Table scores have an id_test, name, score_grammar, score_listening, score_reading, score_final, status_test, status_listening, status_grammar and status_reading.

4. **Table Grammar_question.** The participant table is used to store all grammar_question data. Table of grammar_questions has a id_question, question, choice_a, choice_b, choice_c, choice_d, answer and status.

5. **Table Listening_question.** The participant table is used to store all listening_question data. Table of listening_question has a id_question, id_audio, question, choice_a, choice_b, choice_c, choice_d, answer and status.

6. **Table Audio.** The participant table is used to store all audio data. Table of audio has a id_audio and name_audio.

7. **Table Reading_question.** The participant table is used to store all reading_question data. Table of reading_question, id_question, id_reading, question, choice_a, choice_b, choice_c, choice_d, answer and status.

8. **Table text_Reading.** The participant table is used to store all text_reading data. Table of text_reading has id_reading and text_reading.

9. **Table Conversion.** The participant table is used to store all converted data. Table of conversion has Id, correct_amount, conversion_listening, conversion_grammar and conversion_reading.
3. Results and discussion
The implementation of this application will discuss the function procedures obtained from the design of the TOEFL Test Application System. System testing aims to determine to extend the success of the system in providing convenience to the TOEFL exam for the user.

3.1 Home Menu
The home menu is the main page view that appears when the user first starts the application. This page will display the opening menu of the application before entering the next menu. On the home page of the TOEFL test application system, there are a number of buttons, such as the home menu button and login menu. The home page display of the TOEFL test application system can be seen in Figure 3.

3.2. Login menu
The login menu page is a page display that can only be accessed by admin, officers, and participants who have been registered as users of the application. Admin, officers, and participants are asked to fill in their username and password to be able to access the application. If the user successfully accesses the system, you can choose the menus that are on the next page.

3.3 Participant Dashboard
The participant's dashboard page is the participant's main page view after logging in the participant. The participant's main menu can only be accessed by participants who have successfully logged in. In the main view of the participants, there are names and surnames of participants who will take the exam. On the main page of the participants there is also a button to start the TOEFL test which functions to enter the next menu to start the TOEFL exam. The main page view of the participants can be seen in Figure 5.
3.4. **Listening Question Page**

The page view of the listening question is the appearance of the first session when participants take the TOEFL exam. On the page the system displays mp3 format audio as well as questions and answer choices. Participants can choose the answer by clicking on one of the radio buttons in options a, b, c and d. If participants want to proceed to the next question, participants can press the next button and if you want to check the previous answer, participants can press the previous button. The appearance of the listening question page can be seen in Figure 6.

![Figure 5. Participant Dashboard](image)

![Figure 6. Listening Question Page](image)

3.5. **Grammar Question Page**

The grammar question page display is the second session display when participants take the TOEFL exam. On the page the system displays questions and answer choices. Participants can choose the answer by clicking on one of the radio buttons in options a, b, c and d. If participants want to proceed to the next question, participants can press the next button and if you want to check the previous answer, participants can press the previous button.

![Figure 7. Grammar Question Page](image)
3.6 Reading Question Page

The display of reading questions is the last session when participants take the TOEFL exam. On the page the system displays reading text, questions and answer choices. Participants can choose the answer by clicking on one of the radio buttons in options a, b, c and d. If participants want to proceed to the next question, participants can press the next button and if you want to check the previous answer, participants can press the previous button. The page view of reading questions can be seen in Figure 9.

![Figure 8. Reading Question Page.](image)

3.7 Implementation Algorithm Linear Congruential Generators (LCG).

The implementation of the Linear Congruential Generators (LCG) algorithm in the application functions so that the questions displayed will be randomized so that each question displayed on each participant is different when starting the test, thus preventing the user from memorizing the questions. Randomization of questions is done using the Linear Congruential Generators algorithm. The source code of the Linear Congruential Generators algorithm for grammar problems is as follows:

```php
$a = 21; $c = 17; $m = 100; $z = 1;
$arr = array();
for ($i=0; $i < 40; $i++) {
    $zz = (($a * $z) + $c) % $m;
    $arr[] = $zz;
    $z = $zz;
}$arr = array_diff($arr, ['0']);
array_push($arr, 100); shuffle($arr);
$data_soal_grammar = array();
$x = 0;
foreach ($arr as $my_arr) {
    $data_soal_grammar[$x] = Grammar::where('id_soal', $my_arr)->get();
    $x++;
} Return view('soal_peserta.soal_grammar', compact('data_soal_grammar'));
```

The Linear Congruential Generators source code in reading questions is as follows:

```php
$a = 11; $c = 7; $m = 10; $z = 1;
$arr = array();
$kode_teks_reading = array();
for ($i=0; $i < 10; $i++) {
    $zz = (($a * $z) + $c) % $m;
    $arr[] = $zz;
    $z = $zz;
}$arr = array_diff($arr, ['0']);
```

shuffle($arr);
for ($y=0;$y<5;$y++) {
    $kode_teks_reading[] = $arr[$y];
}
$data_soal_reading = array(array());
$x = 0;
foreach ($kode_teks_reading as $kode_teks) {
    $data_soal_reading[$x] = Reading::where('id_reading',$kode_teks)-
        orderBy('id_soal','asc')
        ->get();
    $x++;
}
return view('soal_peserta.soal_reading',
    compact('data_soal_reading','kode_teks_reading'));

3.8 Implementation Algorithm Linear Congruential Generators (LCG).
Proof of the Linear Congruential Generators (LCG) algorithm in reading questions if proven by
manual calculations are as follows:

Given : a = 11, c = 7, m = 10, Z1 = 1
Formula: Zn = (a . Zn-1 + c) mod m
Answer:
Z1 = (11 x 1 + 7) mod 10
   18 : 10 = 1
   10 x 1 = 10
   18 – 10 = 8
Z2 = (11 x 8 + 7) mod 10
   95 – 90 = 5
Z3 = (11 x 5 + 7) mod 10
   62 – 60 = 2
Z4 = (11 x 2 + 7) mod 10
   29 – 20 = 9
Z5 = (11 x 9 + 7) mod 10
   106 – 100 = 6
Z6 = (11 x 6 + 7) mod 10
   73 – 70 = 3
Z7 = (11 x 3 + 7) mod 10
   40 – 40 = 0
Z8 = (11 x 0 + 7) mod 10
   7 – 0 = 7
Z9 = (11 x 7 + 7) mod 10
   84 – 80 = 4
Z10 = (11 x 4 + 7) mod 10
   51 – 50 = 1
If Z11 is searched, the results are as follows:
Z11 = (11 x 1 + 7) mod 10
   18: 10 = 1
   10 x 1 = 10
   18 – 10 = 8

The results of the Z11 search will be the same as the Z1 search results because it is known that the
number that is randomized is only 10 repetitions.
4. Conclusion

Based on the testing of the algorithm that has been done, the Linear Congruential Generator (LCG) algorithm in this application can be applied correctly to randomize the questions. Calculation of scores on this application system is based on tables of Longman Preparation Course conversion for the TOEFL TEST.

5. References

[1] Aisa, S. (2015). Perancangan Aplikasi Simulasi TOEFL Berbasis Android. Konferensi Nasional Sistem & Informatika.

[2] Dharma, M. P. (t.thn.). Pseudo Random Generator Untuk Kode Acak Pin Pada Sistem Informasi PMDK Online.

[3] Fatansyah.1999. Basis Data. Informatika Bandung, Bandung.

[4] HD. Brown. Language Assessment: Principle and Classroom Practices. New York: Pearson Education, 2004.

[5] Mulyana. (2012). Sistem Informasi TOEFL Berbasis Web Pada Laboratorium Bahasa Universitas AlMuslim.

[6] Munanda, R. (2014). Rancang Bangun Aplikasi Tes TOEFL Pada Android Berbasis Client Server. Lhokseumawe.

[7] Jogiyanto HM. 2000. Analisis dan Desain Sistem Informasi: Pendekatan terstruktur teori dan praktis aplikasi bisnis. Andi. Yogyakarta.

[8] Kendall, K.E dan Kendall, J.E. 2003. Analisis dan Perancangan Sistem.Prehallindo. Jakarta.

[9] Parwanto, T. (2011). Perancangan Aplikasi Simulasi Toefl (Test Of English As Foreign Language). Jakarta.

[10] Phillips, Deborah. Longman Preparation Course for the TOEFL Test, Second Edition. Longman, 10 Bank Street, White Plains, N.Y.

[11] Putra, A. D. (2009). Rancang Bangun Media Pembelajaran TOEFL Berbasis Web.ISBN.

[12] Ramadhan. (2007). Pengembangan Random Number Generator dengan Video dan Suara.

[13] Putranto. (2010). Linear Congruent Method Jenis Pseudo Random Number Generator (PRNG).

[14] Veronika, N. D. (2015). Rancang Bangun Aplikasi Tes TOEFL Menggunakan Algoritma Quick Sort Berbasis Komputer. Jurnal Pseudocode.