Performance Comparison of Linear Congruent Method and Fisher-Yates Shuffle for Data Randomization

Fransiskus Panca Juniawan¹, Harrizki Arie Pradana², Laurentinus³, and Dwi Yun Yuliana Sylfania⁴
¹,²,³,⁴ Department of Computer Science, STMIK Atma Luhur, Jl. Jend. Sudirman, Pangkalpinang, Indonesia
E-mail: ¹fransiskus.pj@atmaluhur.ac.id, ²harrizkiariep@atmaluhur.ac.id, ³laurentinus@atmaluhur.ac.id, ⁴dysylfania@atmaluhur.ac.id

Abstract. Linear Congruent Method (LCM) and Fisher-Yates Shuffle Algorithms are widely known and used in various fields and needs. Both of these algorithms are used because of their advantage and are used for data randomization needs. There have been many studies using these algorithms, but it is not known which algorithm performance is better. In this study, performance comparison testing was carried out in the form of speed testing based on the amount of randomized data. The amount of randomized data varies from 10 to 25,000 data. The tests performed are displayed in tables and graphs. The results obtained, Fisher-Yates algorithm 11.768% faster than LCM.

1. Introduction

Nowadays, the use of computers has been widely applied in all fields of human life. Many algorithms have also been developed that are used to facilitate and help the use of a computer system. In the case of data randomization, there are many algorithms that can be used. Currently, the most popular algorithm with randomization functions are the Fisher-Yates Shuffle and Linear Congruent Method (LCM) algorithms. Both algorithms are known for their advantages.

The Fisher-Yates Shuffle algorithm has been applied to various methods and case studies that use the randomization function [1]-[2]-[3]-[4]. The advantage of this algorithm is the effectiveness of the randomization method and the optimal complexity of the algorithm, likely O(n). In addition, this algorithm is widely used because the randomization of the resulting data will not be the same. So, it avoids repetition and duplication [5]. Another advantage of Fisher-Yates is that this algorithm produces unbiased results. Each permutation of an array has the same possibilities [1]. But this algorithm also has weaknesses, namely the sorting method in the form of sequential sorting [6].

Linear Congruent Method (LCM) is also an algorithm that can be used for randomization of problems and has been known for its easy and fast random number generation function [7]. This method can also be used as an alternative to randomize questions. But the weakness of this method is that there are still the same numbers in each randomization process [8]. At this time, a variation method was used to generate random numbers [9]-[10]-[11].

From the background described above, it is known that both algorithms have been widely applied in various cases with various objectives. However, the performance of the randomization process of the two algorithms is unknown yet. So in this study, the performance of the two algorithms are measured in...
terms of speed and amount of randomized data. After knowing which performance is better, so in the next work can be used the right algorithm in randomization. Besides, this research can be used as a basic, so that in the future it can be improved again.

2. Related Work
The study [12] introduces a new algorithm called MERGESHUFFLE, which is an extreme algorithm that is efficient in generating random permutations or performing random array permutations. This research also proposes a new algorithm called BALANCEDSHUFFLE to further reduce the number of random bits consumed. The study also says that the Fisher-Yates Shuffle can be developed in two ways, namely the algorithm's initial assumptions that allow for discrete uniform variables, and also with the advent of large core clusters and GPUs, there is an interest in making parallel versions of this algorithm.

Another research produces file encryption method using Fisher-Yates shuffle where this algorithm operates on a file consisting of integers that are limited by spaces. Also proposed a new encryption and decryption method that produces an original file containing an integer in it [13].

In 2016 a study was done that is modification of the encryption technique proposed by [12]. This study resulted an image encryption by Blockwise Pixel Shuffling using the Fisher-Yates Shuffling algorithm with pseudorandom permutations that used several security analyses so that it was considered satisfactory [14].

Furthermore, there is a study that proposes algorithm modification with the name Fisher-Yates Chaotic Shuffling algorithm which is used in grayscale image encryption using the wavelet algorithm domain. The proposed algorithm is tested on several standard images. The test results show that the images produced are consistent, safe and efficient. This is intended to achieve safe transmission of grayscale images and can be shared via unsafe networks [15]. The research [16] produces random number generation using Linear Congruent Method to determine a unique pseudonym. Then, in a subsequent study, Linear Congruent Generator (LCG) is used as a random number generator in the Pseudo-Random Generator in cryptographic applications. This algorithm is based on the control distribution of generating random numbers with chaotic phenon congruentia generators [17].

A simpler new algorithm is proposed to generate pseudorandom numbers. The time complexity of the proposed algorithm produces lower values than popular algorithms. Predictability of the generated numbers can be reduced considerably at the cost of increasing the time of primality testing [18]. Next research conducted a comparison of One-time pad random key on the Linear Congruential Generator (LCG) and Quadratic Congruential Generator (QCG). From the experiments that have been carried out it is known that the LCG process is faster than QCG [19].

3. Research Problem
Based on several literature studies that have been described, many users are using the Fisher-Yates Shuffle algorithm and LCM to perform data randomization. However, at this time it is not known which algorithm have the faster randomization process, both in terms of the speed of the data randomization process and in terms of the large number of randomized data. For this reason, several testings were carried out to determine the performance of the two algorithms.

4. Proposed Methodology
Calculation of performance comparison is done by sorting the basic calculations of each algorithm. The first step is to calculate the LCM algorithm, then next step is proceeding the calculation of the Fisher-Yates Shuffle algorithm. To get the test results, the two algorithms are calculated using PHP.

For generating random numbers on the LCM algorithm, pseudorandom number is obtained from the following formula:

\[ N_{j+1} = \left( A \times N_j + B \right) \mod M \]  

(1)

Where:
\( N_j \) = random number to \( j \)
A and B = LCM constant
M = maximum number of random numbers

Then for the Fisher Yates-Shuffle algorithm using the S array randomization function of n elements consisting of numbers 1..n, the calculation is done using the following formula [1]:

For I \leftarrow n \text{ to } 1 \text{ do}
J \leftarrow \text{random integer with } 1
\text{Exchange } (S[j], S[i])
end

4.1. Linier Congruent Method (LCM) Calculation
The detailed steps of the Linear Congruent Method calculation are as follows:

- Initialization Variable $a$ is a prime number where $0 < a < m$
  $a=11$;
- Initialization Variable $c$ is a prime number where $0 < c < m$
  $c=7$;
- Initialization Variable $m$ is the amount of data
  $m=1000$;
- The $X0$ variable is a random value for $X0$ with a range of 0 to 100
  $xn = \text{rand}(0,100);$
  $show =100$;
- Looping as many as the number of variables shown
  if($m$){
    for($i=1; i <= show; i++$)
  }
- Linear Congruent Method Calculation
  $x = \text{metode}(a, xn, c, m, arr);$;
- Put the calculation results into the array
  $x = array\_push(arr, x);$;
- Search for random values through the function method with parameters $a$, $xn$, $c$, $m$, and $arr$
  function metode($a$, $n$, $c$, $rn$, $arr$)
  $x = ((a * n) + c) \mod rn$;
  if(in_array($x$, $arr$));
  metode($a$, $n$, $c$, $rn$, $arr$);
  else
  return $x$;

This LCM method is used to generate random numbers $r1, r2, ..., rn$ which are worth $[0, m]$ by utilizing the previous value. To generate random numbers to $n+1$ ($rn+1$) using LCM method. Determined $a = 4$, $c = 1$ and $r1 = 3$, then random numbers 0 to 8 ($m = 9$) can be calculated.

$r2=((4)(3)+1) \mod 9 = 4$
$r3=((4)(4)+1) \mod 9 = 8$
$r4=((4)(8)+1) \mod 9 = 6$
- Perform randomization using the formulas: $((a * xn) + c) \mod m$

4.2. Fisher-Yates Shuffle Calculation
The detailed steps of the Fisher-Yates Shuffle calculation are as follows:
- Initialization variable count or amount of data = 100
• Perform looping and enter loop values into the data array
  
  ```php
  $value = array();
  $count = 100;
  $data = array();
  for ($y = 0; $y < $count; $y++)
    Array_push($data, $y);
  ```

• Perform looping a number of arrays count
  
  ```php
  $index = 0;
  for ($i = $count - 1; $i > 0; $i--)
    $j = fisher_yates($i, $data, $value);
    $tmp = $data[$i];
    $data[$i] = $data[$j];
    $data[$j] = $tmp;
    Array_push($value, $tmp);
  ```

• Looking for a random value from the i-loop in the fisher_yates function with parameters i, data and value.

• Function fisher_yates searches random values using the formula: (i + 1), 0 ≤ j ≤ i

  ```php
  function fisher_yates($ii, $data, $value)
  $index = rand() % ($ii + 1);
  if (!in_array ($index, $value))
    Avoiding the same number appears again, check the value of array index.
    return $index;
  else
    fisher_yates($ii, $data, $value);
  ```

• Do the iteration based on loop

5. Discussion

For the testing, we input the data with varying amounts. The more data, then randomization process will also be longer. After being applied to the system, the results are shown in table 1 and figure 1.

| Amount of Data | LCM (second) | Fisher-Yates (second) | Percentage (%) |
|----------------|--------------|-----------------------|----------------|
| 10             | 0.003194     | 0.003004              | 5.95           |
| 100            | 0.034499     | 0.030307              | 12.12          |
| 1000           | 0.188667     | 0.146749              | 22.22          |
| 10000          | 1.612128     | 1.420260              | 11.90          |
| 25000          | 11.000212    | 10.268512             | 6.65           |
| Average        |              |                       | 11.768         |
To find out the performance of these algorithms, a number of tests are carried out. The tests carried out are refers to how fast the algorithm do the randomization process based on the amount of data provided. The amount of data performed varies from 10 data, 100 data, 1000 data, 10000 data, and 25000 data. Figure 2 shows the results of the test with 10 amount data. In this test, the results of LCM randomization are 0.003194 second and Fisher-Yates 0.003004 second. So, Fisher-Yates is 5.95% faster. Figure 3 shows the test results with a total of 100 data. The result was 0.034499 second for LCM and 0.030307 second for Fisher Yates. In this test Fisher Yates is 12.15% faster.
Figure 4 is the result of testing 1000 data. The results are LCM randomization required 0.188667 seconds and Fisher-Yates needed 0.146749 seconds, so Fisher-Yates was 22.22% faster. Further testing with 10000 data with Fisher Yates results is 11.9% faster. The details are 1.612128 seconds for LCM and 1.42026 seconds for Fisher Yates. This is shown in Figure 5. The last test is using 25,000 amount data. From this test the results obtained are LCM takes 11,000212 seconds while Fisher-Yates takes 10.268512 seconds. This means that Fisher-Yates is 6.65% faster.
6. Conclusion
In this work, two data randomization algorithms are compared. Comparison is done to determine the performance between the two algorithms. By knowing the performance of each algorithm, so in the next work can be used the faster algorithm in randomization.

Based on the test results it can be concluded that in average the Fisher-Yates algorithm has a faster processing time than the LCM, which is worth 11.768%. This is directly proportional to the testing of the amount of data carried out ranging from 10, 100, 1000, 10000, and 25000 data. For data of 10, Fisher-Yates is more efficient with 5.95% efficiency. While for data totalling 100 Fisher-Yates was 12.12% faster. For 1000 data Fisher Yates was 22.22% faster. For data totalling 10000 obtained results of 11.90% faster for Fisher-Yates. The last, for data totalling 25,000 Fisher Yates is 6.65% faster.

7. Acknowledgement
Thanks to the source of the STMIK Atma Luhur internal research fund for the 2017/2018, so this research can be carried out.

8. References
[1] Saeed, S., Umar, M.S., Ali, M.A., & Ahmad, M. (2014). Fisher-Yates Chaotic Shuffling Based Image Encryption. International Journal of Information Processing, 8(3), 31-41.
[2] Nithiya, D.G., Sharmila. S., Saranya, N., Rajkumar. K.K., & Gomathi. (2016). Novel Architecture for Data – Shuffling Using Enhanced Fisher Yates Shuffle Algorithm. International Journal of Engineering Science and Computing, 6(5), 4932-4935.
[3] Aishwarya, C., & Beny, J.R. (2015). Novel Architecture for Data – Shuffling Using Fisher Yates Shuffle Algorithm. International Journal of Scientific Research in Science, Engineering and Technology, 1(6), 387-390.
[4] Sarmah, B.K., & Chakrabarty, D. (2014). Testing of Randomness of the Number Generated by Fisher and Yates. International Journal of Engineering Sciences & Research Technology, 3(11), 632-636.
[5] Haditama, I., Slamet, C., Rahman, D.F. (2016). Implementasi Algoritma Fisher Yates dan Fuzzy Tsukamoto Dalam Game Kuis Tebak Nada Sunda Berbasis Android. Jurnal Online Informatika, 1(1), 51-58.
[6] Kiselyov, O. (3 Sep 2001). Provably perfect shuffle algorithms. Retrieved 2018-07-31.
[7] Alhomdy, S.A., Abdullah, S.N., & Algabri, M.N. (2016). Generating Random Data using 3 Nonlinear Functions. International Journal of Computer Applications, 152(4), 6-10.
[8] Limbong, T., & Simarmata, J. (2015). Implementasi Linear Congruent Method (LCM) Untuk Pengacakan Soal Ujian Berkategori, Proceedings Seminar Nasional Ilmu Komputer: Vol. 26. Managing information and technology (pp. 96–100). Medford, NJ: Learned Information.
[9] Yajie, L., & Mingyue, Z. (2016). Pseudo Random Number Generating Based on Linear Congruence Algorithm. Journal of Residuals Science & Technology, 13(5), 651-653.
[10] Agarwal, R., & Agarwal, G. (2016). An Efficient Method of Generating Random Numbers from Congruence Equations for Cryptographic Applications. International Journal of Computer Science Engineering and Technology, 6(7), 290-293.
[11] Jeong, J., & Kim, J.S. (2016). Automatic Fortified Password Generator System Using Special Characters. International Journal of Fuzzy Logic and Intelligent Systems, 15(4), 295-299.
[12] Bacher, A., Bodini, O., Hollender, A., & Lumbroso, J.O. (2015). MergeShuffle: A Very Fast, Parallel Random Permutation Algorithm. CoRR, abs/1508.03167.
[13] Hazra, T.P., Ghosh, R., Kumar, S., Dutta, S., & Chakraborty, A.K. (2015). File Encryption using Fisher-Yates Shuffle. International Conference and Workshop on Computing and Communication (IEMCON): Vol. 6. (pp. 1-7). Canada: University of British Columbia.
[14] Hazra, T.P., & Bhattacharyya, S. (2016). Image Encryption by Blockwise Pixel Shuffling Using Modified Fisher Yates Shuffle and Pseudorandom Permutations. IEEE Annual Information
Technology, Electronics and Mobile Communication Conference (IEMCON): Vol. 7. (pp. 1-6). Canada: University of British Columbia.

[15] Saeed, S., Umar, M.S., Ali, M.A., & Ahmad, M. (2014). A Gray-Scale Image Encryption Using Fisher-Yates Chaotic Shuffling in Wavelet Domain. IEEE International Conference on Recent Advances and Innovations in Engineering (ICRAIE): Vol. 1. (pp. 1-5). India: Poornima College of Engg, Poornima University.

[16] Sun, G., Xie, Y., Liao, D., Yu, H., & Chang, V. (2016). User-Defined Privacy Location-Sharing System in Mobile Online Social Networks. Journal of Network and Computer Applications, 86, 34-45.

[17] Vajargah, B.F., & Asghari, R. (2016). A Novel Pseudo-Random Number Generator for Cryptographic Applications. Indian Journal of Science and Technology, 9(6), 1-5.

[18] Das, S., Dasgupta, P., Pandey, A., & Roy, P. (2015). A close encounter with Random Numbers. Proceedings of the 2015 Third International Conference on Computer, Communication, Control and Information Technology (C3IT): Vol. 3. (pp. 1-6). India: IEEE.

[19] Apdilah, D., Harahap, M.K., Khairina, N., Husein, A.M., & Harahap, M. (2018). A Comparison of One Time Pad Random Key Generation using Linear Congruential Generator and Quadratic Congruential Generator. Journal of Physics Conf. Series 1007 (2018), 1-6.