A Tutorial on New Methods and Algorithms for Solving LCM and GCD

M. SABRIGIRIRAJ
Department of Information Technology,
Hindusthan College of Engineering and Technology, Coimbatore,
INDIA

Abstract: - This paper presents a new tutorial on the innovative methods and algorithms for solving the commonly used mathematical problems namely Least Common Multiple (LCM) and Greatest Common Divisor (GCD). Starting with the basic concepts, the tutorial takes the readers through various new techniques with step-by-step example implementations for both LCM and GCD. Appropriate pseudocodes for the proposed techniques based on iterative and Euclidean approaches are presented towards effective problem-solving. The pseudocodes presented are focused on using functions and customized to solve problems with larger size inputs and thereby facilitate scalability. A performance comparison study is also done to evaluate the efficiency of the proposed methods. This tutorial article provides a solid base for readers aspiring to master the art of problem-solving and programming with a systematic approach.

Key-Words: - LCM, GCD, Programming, Algorithms, Pseudocode, Problem-Solving.

Received: October 15, 2023. Revised: May 15, 2024. Accepted: May 17, 2024. Published: June 20, 2024.

1 Introduction

LCM (Least Common Multiple) and GCD (Greatest Common Divisor) are two basic mathematical operations that find extensive applications in Engineering, Science, Management, and Technology. In particular, LCM and GCD are employed during the optimization of resources in several critical applications ranging from electrical motor design, cloud computing, cyber-physical systems, and radar engineering to the Industrial Internet of things. The problem of LCM and GCD finds extensive attention from researchers in various perspectives ranging from focusing on improved teaching and learning outcomes to applying in a variety of domains focusing on real-world applications.

Researchers have given enormous attention to studying and exploring the concepts of LCM and GCD over the past years. Research on the verification of sentences involving LCM is explored in [1]. A detailed examination of the necessity of computational thinking in primary mathematics education, along with ways to integrate it into the teaching and learning process, is provided in [2]. [3], presents an interactive computer-based learning approach for LCM and GCD using animations. [4], focuses on the development of an educational game media based on an Android application for effective learning of LCM and GCD. The importance of developing a mathematics textbook using realistic teaching methods to motivate elementary school students is discussed in [5]. Problem-based learning of LCM and GCD is shown to be more effective and attractive for school students, [6]. A report on the Indonesian Realistic Mathematics Education approach, [7] and the safari numbered model, [8], demonstrates a positive effect on students' learning outcomes regarding GCD and LCM in primaryschool. A study [9], [10], presents the broader picture of computers drastically changing mathematics learning and enhancing the potential of computational thinking in formal school settings. The primary building blocks of teaching LCM and GCD are disseminated in [11]. [12], successfully demonstrates a gaming technique for teaching LCM and GCD to improve learning outcomes.

Research [13], delves into a method to process repetition in musical composition using LCM, highlighting the close relationship between mathematics and music. [14], exhibits finding LCM and GCD using DNA computing. A realistic mathematical approach, [15], [16], [17], [18] is employed to teach LCM and GCD, increasing students' reasoning abilities. Computational thinking increases while solving LCM and GCD with Scratch, as demonstrated in [19]. [20], introduces a digital creative magic e-book for learning mathematical concepts, including LCM and GCD, designed and adopted for school students. New teaching methods for LCM and GCD are researched with a focus on their meanings, [21]. Proactive interference is noted in [22], when students use LCM strategies to solve GCD problems.

The effectiveness of a mathematics teacher's actions prompting various thoughts is examined in [23], while teaching LCM and GCD. The underlying issues for misconceptions among students during the computation of LCM and GCD are analyzed in [24], [25]. Common errors made by students while solving LCM and GCD, along with other learning obstacles and troubleshooting tips, are analyzed in [26], [27], [28], [29]. [30], employs three different visual models to teach LCM and GCD, conducting a comparative study. A set of real problems involving LCM and GCD in practice is demonstrated in [31]. The application of LCM and GCD for computing distractor suites in developing choices for a multiple-choice question is demonstrated in [32]. LCM and GCD are used as sample mathematical concepts to assess the performance of the ChatGPT-4 code interpreter, [33]. They also find a place in the performance evaluation of a multispace model, [34], along with network security and cryptography techniques, [35], [36].

Although a vast literature is available concentrating on teaching-learning methods for LCM and GCD using books, eBooks, interactive mode learning techniques, gaming methods, problem-solving techniques, magic books, animation software, in addition to ways of tracing misconceptions and troubleshooting methods, still there is a scope to expand it from a programming point of view. Expanding the teaching and learning methods of LCM and GCD from a programming perspective increases the computational thinking, logical thinking, project management, and problem solving capabilities of the readers. To address this research gap, the authors have explored and introduced innovative techniques from a programming standpoint. In this paper, the concepts of LCM and GCD have been dealt with through simple straightforward examples, and new methods have been suggested from a programming perspective. Additionally, a performance comparison study is conducted among the proposed techniques.

The rest of this article is prepared as follows: Sections 2 through 3 illustrate LCM and GCD through simple examples and new methods and algorithms are proposed with appropriate pseudocodes. Performance comparison between the proposed techniques is carried out in section 4. Section 5 concludes the paper and points out future research directions.

2 Least Common Multiple (LCM)

Least Common Multiple (LCM) is a mathematical technique to find the smallest common multiple for a set of given numbers. It is very trivial to note that the lowest common multiple may lie anywhere between the biggest input (among the given set of numbers) and the product of all the given numbers.

Example 1. Find the LCM of a) 4 and 6 b) 4 and 5 c) 5 and 7 d) 2 and 6. Write a pseudocode for the same.
Solution:

a) LCM of 4 and 6 needs to be a number between 6 and 24. Then, consider the series 6, 7, 8, 9, …, 24. In this series, 12 is the smallest number that divides both inputs 4 and 6 without leaving any remainder. So, LCM is 12.
b) 20
c) 35
d) 6

Algorithm LCM_1 (a, b)

```
{
Max = maximum (a, b)
while ((Max % a ≠ 0) 'OR' (Max % b ≠ 0)
        Max +=1
return (Max)
}
```

The above pseudocode Algorithm LCM_1 (a, b) may be easily extended for any number of inputs and it is very trivial.

An alternative technique to find LCM is to begin by comparing one's multiple of all the given inputs. When the one's multiple of the inputs is not equal, the multiplying integer values on the smaller sides are incremented from their current value by 1 and the comparison is again done. This process is repeated again and again until all sides become equal. This is illustrated below in Example 2.

Example 2: Find the LCM of 4 and 6. Write a pseudocode for the same.
Solution:
$4 * 1 < 6 * 1$
$4 * 2 > 6 * 1$
$4 * 2 < 6 * 2$
$4 * 3 = 6 * 2$
$12 = 12$

Algorithm LCM_2 (a, b)

```
{
Input a, b
Initialize i = j =1
```

```
while (a*i ≠ b*j)
        if (a*i > b*j)
          j ++
        else
          i ++
return (a*i)
}
```

Example 3: Find the LCM of 2, 3, 4. Write a pseudocode for the same.

$2 * 1 < 3 * 1 < 4 * 1$
$2 * 2 < 3 * 2 > 4 * 1$
$2 * 3 = 3 * 2 < 4 * 2$
$2 * 4 < 3 * 3 > 4 * 2$
$2 * 5 > 3 * 3 < 4 * 3$
$2 * 6 = 3 * 4 = 4 * 3$
$12 = 12 = 12$

Algorithm LCM_3 (a, b, c)

```
{
Input a, b, c,
Initialize i = j = k = 1
while (a*i ≠ b*j ≠ c*k)
        if (a*i > b*j) 'AND' (a*i > c*k)
          j ++, k ++
        else if ( b*j > a*i) 'AND' (b*j > c*k)
          i ++, k ++
        else if (c*k > a*i) 'AND' (c*k > b*j)
          i ++, j ++
        else if (a*i > b*j) 'AND' (a*i = = c*k)
          j ++
         else if (b*j > a*i) 'AND' (b*j = = c*k)
          i ++
        else
          k ++
return (a*i)
}
```

The above pseudocodes, Algorithm LCM_2 (a, b) and Algorithm LCM_3 (a, b, c) may be extended for any number of inputs, but the number of 'else if' conditions will exponentially increase (and so code length increases) as the input size increases. However, the concept of '*function call'* may be employed to find the LCM of a bigger size input set because LCM (a, b, c) = LCM (LCM (a, b), c). Here, the code length remains the same irrespective of the input size. The pseudocode for a set of N numbers stored as an array or list is given below.

Algorithm LCM_N (A [1…N])

```
{
K = A [1]
for (i = 2; i ≤ N; i ++)
        K= LCM_1 (K, A[i])
return (K)
}
```

It can be easily deduced that the time complexity for finding the LCM of N given numbers is O(N*K). Also, the time complexity of finding the LCM of two input numbers is O(K) and K denotes the approximate number of comparisons involved.

3 Greatest Common Divisor (GCD)

GCD (also known as the highest common factor or HCF) is the biggest positive integer that divides the given set of positive integers exactly without leaving any remainder. It is very trivial to note that any common divisor for the given set of input must lie anywhere between the smallest input (among the given set of input numbers) and unity.

Example 4. Find the GCD of a) 4 and 6 b) 4 and 5 c) 5 and 7 d) 2 and 6. Write a pseudocode for the same.

Solution:

a) GCD of 4 and 6 need to be a number between 4 and 1. Then, consider the series 4,3,2,1. In this series, 2 is the biggest number that divides both inputs 4 and 6 without leaving any remainder. So, GCD is 2.
b) 1
c) 1
d) 2

Algorithm GCD_1 (a, b)

```
{
Min = minimum (a, b)
while ((a % Min  ≠ 0) 'OR' (b % Min  ≠ 0))
          Min - = 1
return (Min)
}
```

The above pseudocode Algorithm GCD_1 (a, b) may be easily extended for any number of inputs and it is very trivial.

An alternative technique to solve the 2-input GCD problem is repeated subtraction, [15]. In the first step, the bigger input number needs to be subtracted the smaller input number and the result to be saved. For each of the subsequent steps, the first biggest number among the three (two input numbers and one saved result of the previous step) needs to be discarded and the second biggest number has to subtract the smallest number and the result to be saved. The above step needs to be repeated again and again till the subtracted result reaches zero. It is important to note that GCD (*m*, 0) = *m*. This is illustrated in the following Example 5.

Example 5: Find the GCD of 12 and 3 by repeated subtraction method. Write a pseudocode for the same.

GCD (12, 3) = GCD (12-3, 3)
GCD (9, 3) = GCD (9-3, 3)
GCD (6, 3) = GCD (6-3, 3)
GCD (3, 3) = GCD (3-3, 3)
GCD (0, 3) = 3

Algorithm GCD_2 (a, b)

```
{
Input a, b
while (a ≠ b)
        if (a > b)
          a = a – b
        else
          b = b - a
return (a)
}
```

This technique can be easily extended for 3 input problems and is illustrated in Example 6.

Example 6: Find the GCD of 12, 36, and 3 by repeated subtraction method. Write a pseudocode for the same.

GCD (12, 36, 3) = GCD (12, 36-12, 3)
GCD (12, 24, 3) = GCD (12, 24-12, 3)
GCD (12, 12, 3) = GCD (12-12, 12, 3)
GCD (0, 12, 3) = GCD (0, 12-3, 3)
GCD (0, 9, 3) = GCD (0, 9-3, 3)
GCD (0, 6, 3) = GCD (0, 6-3, 3)
GCD (0, 3, 3) = GCD (0, 3-3, 3)
GCD (0, 0, 3) = 3

Algorithm GCD_3 (a, b, c)

```
{
Input a, b, c
while ( (a = b ≠ 0) 'OR' (b = c ≠ 0) 'OR' (a = c ≠ 0) )
        if (a > b) and (a > c) and (b > c)
          a = a – b
        elseif (a > b) and (a > c) and (b <= c)
          a = a – c
        elseif (b > a) and (b > c) and (a > c)
          b = b - a
        elseif (b > a) and (b > c) and (a <= c)
          b = b - c
        elseif (c > a) and (c > b) and (a > b)
          c = c – a
        elseif (c > a) and (c > b) and (a <= b)
          c = c – b
        elseif (a == b) and (a < c)
          a = a – b
        elseif (b == c) and (b < a)
          b = b – c
        else
          c = c – a
return (maximum (a, b, c))
}
```

The above pseudocodes, Algorithm GCD_2 (a, b) and Algorithm GCD_3 (a, b, c) may be extended for any number of inputs, but the number of 'else if' conditions will exponentially increase (and so code length increases) as the input size increases. However, the concept of '*function call*' may be employed to find the GCD of a bigger size input set because GCD (*a*, *b*, *c*) = GCD (GCD (*a*, *b*), *c*). Here, the code length remains the same irrespective of the input size.

The pseudocode for a set of N numbers stored as an array or list is given below.

Algorithm GCD_N (A [1…N])

```
{
K = A [1]
for (i = 2; i ≤ N; i ++)
        K= GCD_1 (K, A[i])
return (K)
}
```

Another alternative technique to solve the 2-input GCD problem is Euclid's method [16]. The basic concept of the Euclidean algorithm is given as follows:
GCD (*m*, *n*) = GCD (*n*, *m* mod *n*)
GCD (*m*, 0) = *m*

Example 7: Find the GCD of 8 and 12 by Euclid's method. Write a pseudocode for the same.
Solution:
GCD (8,12) = GCD (12, 8)
GCD (12, 8) = GCD (8, 4)
GCD (8, 4) = GCD (4, 0)
GCD (4, 0) = 0
Hence, GCD (8, 12) = 4

Algorithm GCD_3 (a, b)

```
{
while (b ≠ 0)
          (a, b) = (b, a % b)
return (a)
}
```

It can be easily deduced that the time complexity for finding the GCD of N given numbers is O(N*K). Also, the time complexity of finding the GCD of two input numbers is O(K) and K denotes the approximate number of comparisons involved.

4 Results and Discussion

A detailed programming study was carried out to evaluate the performance of the proposed techniques for LCM and GCD in terms of their execution time. The evaluation of the proposed techniques was carried out using an HP Laptop configured with a core i3 processor with 8GB RAM and 512 MB SSD. Coding was done using Python programming language. The inputs for finding LCM and GCD were randomly generated using the built-in *random.randint()* function in python. Input size was varied between 2 to 25 for the evaluation study. To evaluate the proposed LCM techniques, function calls were made to LCM_1(a, b) and LCM_2(a, b). With the same inputs, function calls were also made to GCD_1(a, b), GCD_2(a, b), and GCD_3(a, b) to evaluate the proposed GCD techniques. Table 1 and Table 2 show the execution time for computing LCM and GCD respectively with the proposed techniques. From Table 1, it can be noted that LCM_2(a, b) executed faster compared to LCM_1(a, b) because LCM_2(a, b) employs fewer iterations than LCM_1(a, b). From Table 2, it can be noted that GCD_3(a, b) executed faster when compared to GCD_1(a, b) and GCD_2(a, b) because GCD_3(a, b) employs fewer iterations than GCD_1(a, b) and GCD_2(a, b). Another noteworthy observation is that program execution time does not increase linearly with input size. It is likely due to the random nature of inputs and also depends on the availability of common factors. In some cases, even if the input size is less, due to the absence of common factors, both LCM and GCD consume more execution time when compared to a higher number of inputs with common factors.

Table 1. Execution time with the proposed LCM techniques

Input Size	Execution time in seconds	
	LCM_1(a, b)	LCM_2(a, b)
2	$1.4543*10^{-5}$	$5.7220*10^{-6}$
3	$6.6757*10^{-6}$	$2.7894*10^{-6}$
4	$1.9311*10^{-5}$	$1.0490*10^{-5}$
5	$1.4305*10^{-5}$	$1.0494*10^{-5}$
6	$1.6450*10^{-5}$	$1.3351*10^{-5}$
7	$2.3126*10^{-5}$	$1.7642*10^{-5}$
8	$5.3644*10^{-5}$	$2.4318*10^{-5}$
9	$1.5258*10^{-5}$	$1.4305*10^{-5}$
10	$2.6941*10^{-5}$	$2.5987*10^{-5}$
15	$3.3140*10^{-5}$	$2.0027*10^{-5}$
20	$2.5749*10^{-5}$	$2.5272*10^{-5}$
25	$8.7976*10^{-5}$	$7.9154*10^{-5}$

Table 2. Execution time with the proposed GCD techniques

Input Size	Execution time in microseconds		
	GCD_1(a, b)	**GCD_2(a, b)**	**GCD_3(a, b)**
2	$1.0967*10^{-5}$	$8.5830*10^{-6}$	$4.2915*10^{-6}$
3	$1.5497*10^{-5}$	$8.3446*10^{-6}$	$2.6226*10^{-6}$
4	$1.6689*10^{-5}$	$1.2159*10^{-5}$	$7.8678*10^{-6}$
5	$2.0265*10^{-5}$	$7.3909*10^{-6}$	$4.7683*10^{-6}$
6	$1.4305*10^{-5}$	$1.2636*10^{-5}$	$9.0599*10^{-6}$
7	$2.0265*10^{-5}$	$1.0251*10^{-5}$	$7.6293*10^{-6}$
8	$1.3113*10^{-5}$	$1.3351*10^{-5}$	$6.6757*10^{-6}$
9	$1.1682*10^{-5}$	$1.1205*10^{-5}$	$9.0599*10^{-6}$
10	$1.6927*10^{-5}$	$1.1682*10^{-5}$	$7.1525*10^{-6}$
15	$4.2676*10^{-5}$	$2.5272*10^{-5}$	$1.7881*10^{-5}$
20	$2.8848*10^{-5}$	$1.8835*10^{-5}$	$1.3828*10^{-5}$
25	$3.6954*10^{-5}$	$2.3365*10^{-5}$	$2.3126*10^{-5}$

5 Conclusion

This tutorial paper revisited LCM and GCD through simple examples and proposed new ways to solve them, along with appropriate pseudocodes. A comparative study was also conducted to evaluate the supremacy of the proposed methods. There exists a huge scope to extend this paper in a multitude of domains. Firstly, the similar study may be conducted to explore other common mathematical operations from a programming perspective. Secondly, LCM and GCD operations may be implemented in VLSI technology and a performance evaluation study to be done. Thirdly, the implementation of LCM and GCD using quantum computing circuits is to be investigated. Fourthly, implementing LCM and GCD using optical computing techniques may be explored. Finally, applications of artificial intelligence and computational techniques with LCM and GCD are an exciting area for exploration.

References:

[1] Fujiwara, Machiko, and Kenzo Iwama, "A program that acquires how to solve problems in mathematics" *WSEAS Transactions on Computers*, Vol.8, No.8, 2009, pp.1337-1347, [Online]. http://www.wseas.us/e-library/transactions/computers/2009/29-610.pdf (Accessed Date: June 06, 2024).

[2] Nordby, Siri Krogh, Annette Hessen Bjerke, and Louise Mifsud, "Computational thinking in the primary mathematics classroom: A systematic review", *Digital Experiences in Mathematics Education*, Vol.8, No.1, 2022, pp.27-49, https://doi.org/10.1007/s40751-022-00102-5.

[3] Jalinus Jalinus, Jesi Alexander Alim, "Development of Interactive Computer Based Media for Learning Mathematics on Greatest Common Divisor (GCD) and Least Common Multiple (LCM) Topics", *Proceedings of the UR International Conference on Educational Sciences*, Pekanbaru, Indonesia,2018, pp.630-638, [Online]. https://ices.prosiding.unri.ac.id/index.php/ICES/article/view/6612 (Accessed Date: June 06, 2024).

[4] Rahmawati I., Ayun, N. Q., Mariana, N., Indrawati, D., Wiryanto, W., Budiyono, B., & Istianah, F., "Edu-Game media based on Android to learn Least Common Multiplication (LCM) and Great Common Divisor (GCD) for the 4th graders", In *Journal of Physics: Conference Series,* Indonesia, IOP Publishing, Vol. 1987, No.1, 2021, pp. 012042. DOI: 10.1088/1742-6596/1987/1/012042.

[5] Alim, Jesi Alexander, Neni Hermita, Melvi Lesmana Alim, Tommy Tanu Wijaya, and Jerito Pereira, "Developing a Math Textbook using realistic Mathematics Education Approach to increase elementary students' learning motivation" *Jurnal Prima Edukasia*, Vol.9, No. 2, 2021, pp 193-201. DOI: 10.21831/jpe.v9i2.39393.

[6] Li, Hui-Chuan, and Tsung-Lung Tsai. "The effects of a problem-based learning intervention on primary students' performance on greatest common factor and least common multiple and on their attitudes towards mathematics", *International Journal of Innovation and Learning*, Vol. 31, No.1, 2022, pp.51-69, https://doi.org/10.1504/IJIL.2022.119636.

[7] Safitri, Nurul Ulfa, and Syamsu Arlis. "The Effect of Indonesian Realistic Mathematics Education Approach On Students' Learning Outcome On GCD And LCM For Fourth Grade Students", *e-Jurnal Inovasi Pembelajaran Sekolah Dasar*, Vol.8, No.1, 2020, [Online]. https://ejournal.unp.ac.id/students/index.php/pgsd/article/view/8098 (Accessed Date: June 06, 2024).

[8] Safari, Ibnu Wahid, Muhammad Darwis, Adi Kurnia, and Rahmat Basuki. "Improving of learning outcomes in greatest common divisor and lowest common multiple mater by developed safari numbered model." *Desimal: Jurnal Matematika,* Vol.6, no. 1, 2023, pp. 99-108.

[9] Ng, Oi-Lam, and Zhihao Cui. "Examining primary students' mathematical problem-solving in a programming context: Towards computationally enhanced mathematics education", *ZDM–Mathematics Education,* Vol. 53, No.4, 2021, pp.847-860, https://doi.org/10.1007/s11858-020-01200-7.

[10] Suarsana, I. Made, "Developing interactive digital mathematics book with multi representation approach for deaf students", *International Journal of Emerging Technologies in Learning,* Vol. 16, No. 13, 2021, pp.128. DOI:10.3991/ijet.v16i13.22459.

[11] Kiss, Anna, "Prime building blocks in the mathematics classroom", *Teaching Mathematics and Computer Science,* Vol.18, No.4, 2020, pp.217-228, https://doi.org/10.5485/TMCS.2020.0493.

[12] Fatimah, Siti, Idawati Idawati, Mardiah Astuti, Indri Mahmudah, and Masnun Baiti. "Effect of Cogklak Game Method to Student's Mathematics Learning Outcome at State Elementary School" *JIP (Jurnal Ilmiah PGMI)* Vol. 7, no. 2, 2021, pp.150-156, https://doi.org/10.19109/jip.v7i2.11089.

[13] Budiawan, Hery, and Rizky Fauzy Ananda, "Music and Mathematics: Processing Repetition Techniques with the Least Common Multiple (LCM) in Rajékan Music Composition", *Harmonia: Journal of Arts Research and Education,* Vol. 23, No.1, 2023, pp. 208-222, https://doi.org/10.15294/harmonia.v23i1.41034.

[14] Sarkar Mayukh, and Prasun Ghosal, "Mathematics using DNA: Performing GCD and LCM on a DNA Computer." *2016 IEEE international symposium on nanoelectronic and information systems (iNIS)*, Gwalior, India, 2016, pp.240-243, DOI: 10.1109/iNIS.2016.062.

[15] Fauzan A., Y. Yerizon, and D. Yulianti, "The RME-based local instructional theory for teaching LCM and GCF in primary school", *Journal of Physics: Conference Series*, Indonesia, IOP Publishing, Vol. 1554, No.1, 2020, DOI: 10.1088/1742-6596/1554/1/012078.

[16] Yulianti, Ditria, and Ahmad Fauzan, "Development of Local Instruction Theory Topics Lowest Common Multiple and Greatest Common Factor Based on Realistic Mathematics Education in Primary Schools", *International Journal of*

Educational Dynamics, Vol.1, No.1, 2018, pp.222-235, https://doi.org/10.24036/ijeds.v1i1.59.

[17] Napitupulu, Rika Cintya, Jesi Alexander Alim, Neni Hermita, and Berdriati Ibrahim. "Development of an RME-Based Hypothetical Learning Trajectory of Least Common Multiple for Elementary School Students." *Journal Of Teaching And Learning In Elementary Education*, Vol. 4, no. 2, 2021, pp.180-191, http://dx.doi.org/10.33578/jtlee.v4i2.7885.

[18] Suyanti, Sri, Rohana Rohana, and Ali Fakhrudin. "Development of indonesian realistic mathematics education-based digital module on mathematic in elementary school", *JIP (Jurnal Ilmiah PGMI)*, Vol.7, No.2, 2021, pp.141-149, https://doi.org/10.19109/jip.v7i2.10557.

[19] Rodríguez-Martínez, José Antonio, José Antonio González-Calero, and José Manuel Sáez-López, "Computational thinking and mathematics using Scratch: an experiment with sixth-grade students", *Interactive Learning Environments*, vol. 28, No.3, 2020, pp.316-327, https://doi.org/10.1080/10494820.2019.1612448.

[20] Faizah, Hanim, Eko Sugandi, and Wahyu Susiloningsih, "Development of mathematics digital creative (Magic) book for elementary school", *Journal of Teaching and Learning in Elementary Education*, Vol.5, No.1, 2022, pp.95-104, http://dx.doi.org/10.33578/jtlee.v5i1.7911.

[21] Pang, JeongSuk, and YuJin Lee. "An Investigation of Teaching Methods of Finding out the Greatest Common Divisor and the Least Common Multiple Focused on Their Meanings", *Journal of Elementary Mathematics Education in Korea*, Vol.22, No.3, 2018, pp.283-308, [Online]. https://koreascience.kr/article/JAKO201831663568645.page (Accessed Date: June 06, 2024)

[22] Lestari, Ana Puji, Erry Hidayanto, and Sukoriyanto Sukoriyanto, "Proactive interference of seventh grade students in solving problems of the greatest common divisor", *AIP Conference Proceedings*, Indonesia, Vol. 2330, No.1, 2021, https://doi.org/10.1063/5.0043382.

[23] Celik, Aytug Ozaltun, and Esra Bukova Guzel, "How to improve a mathematics teacher's ways of triggering and considering divergent thoughts through lesson study," *International Electronic Journal of Mathematics Education,* Vol.15, No.3, 2020, https://doi.org/10.29333/iejme/8461.

[24] Sutarto, Intan Dwi Hastuti, Tomi Listiawan, Sutopo, Aan Komariah, Mohammadreza Dabirnia, "Fourth-Grade Primary School Students' Misconception on Greatest Common Factor and Least Common Multiple", *Education Research International*, Vol. 2021, 2021, pp. 1-11, https://doi.org/10.1155/2021/6581653.

[25] Anwar, Khoirul,"Misconceptions of Students Solving Mathematics Word Problem in Material the Least common Multiple and Greatest Common Factor", Diss. Universitas Islam Negeri Maulana Malik Ibrahim, 2018. http://etheses.uin-malang.ac.id/12389/ (Accessed Date: June 06, 2024)

[26] Lestari, Ana Puji, "Analysis of Students' Errors in Solving the Least Common Multiple and Greatest Common Divisor Word Problems", *Eduma: Mathematics Education Learning*, Vol.11, No.2, 2022, pp. 191-202. DOI: 10.24235/eduma.v11i2.9324.

[27] Martinez, Silvia, and Jose C. Valverde, "Influence of Context on Greatest Common Divisor Problem Solving: A Qualitative Study", *Mathematics*, Vol.10, No.8, 2022, https://doi.org/10.3390/math10081325.

[28] Pratiwi, Inne Marthyane. "Learning obstacles analysis of lowest common multiple and greatest common factor in primary school." *Jurnal Elemen*, Vol. 9, no. 2, 2023, pp. 440-449, https:/doi.org/10.29408/jel.v9i2.12359.

[29] Siti Muniroh, Siti Maghfirotun Amin, Tatag Yuli Eko Siswono, "Training Model of Troubleshooting Mathematics Materials of LCM And GCD Through PBL", *Proceedings of the 2nd International Conference on Education Innovation (ICEI 2018)*, Indonesia, 2018, pp.180-184. doi: 10.2991/icei-18.2018.39.

[30] Lipovača, Karmelita, and Edin Liđan, "Evaluation of three visual models for the greatest common factor and the least common multiple", *Post Scriptum*, 2020, pp. 7-19, [Online]. https://www.ceeol.com/search/article-detail?id=869958 (Accessed Date: June 06, 2024).

[31] Giang, Hoang Minh, Nguyen Thi Thu Hoa, and Nguyen Thi Hong, "Some real problems and application of the greatest common divisor, the least common multiple", *Scientific*

Journal of Hanoi metropolitan university, vol.62, 2022, [Online]. https://vjol.info.vn/index.php/otn1/article/download/82442/70270/ (Accessed Date: June 06, 2024).

[32] Kosh, Audra E., "Distractor suites: A method for developing answer choices in automatically generated multiple-choice items", *Journal of Applied Testing Technology,* Vol.22, 2021, pp. 12-22, [Online]. https://jattjournal.net/index.php/atp/article/vie5880 (Accessed Date: June 06, 2024).

[33] Zhou, A., Wang, K., Lu, Z., Shi, W., Luo, S., Qin, Z., Lu, S., Jia, A., Song, L., Zhan, M. and Li, H., "Solving challenging math word problems using gpt-4 code interpreter with code-based self-verification", *arXiv preprint arXiv:2308.07921*, 2023, https://doi.org/10.48550/arXiv.2308.07921.

[34] Khairuddin, Taufiq Khairi Ahmad. "Some mathematical descriptions of multi-connected system of fuzzy state space model via number theory approach", *WSEAS Transactions on Systems,* Vol. 10, No.9, 2011, pp. 271-280, [Online]. http://www.wseas.us/e-library/transactions/systems/2011/53-578.pdf (Accessed Date: June 06, 2024).

[35] Beletsky, Anatoly, "Factorization of the Degree of Semisimple Polynomials over the Galois Fields of Arbitrary Characteristics", *WSEAS Transactions on Mathematics*, vol.21, 2022, pp, 160-172, https://doi.org/10.37394/23206.2022.21.23.

[36] Kumari, Munesh, Kalika Prasad, Bahar Kuloğlu, and Engin Özkan. "The k-Fibonacci group and periods of the k-step Fibonacci sequences." *WSEAS Transactions on Mathematics, Vol.* 21, 2022, pp. 838-843, https://doi.org/10.37394/23206.2022.21.95.

Contribution of Individual Authors to the Creation of a Scientific Article (Ghostwriting Policy)

The author solely contributed in the present research, at all stages from the formulation of the problem to the final findings and solution.

Sources of Funding for Research Presented in a Scientific Article or Scientific Article Itself

No funding was received for conducting this study.

Conflict of Interest

The author has no conflicts of interest to declare.