

## Article

# Math Can Be Visual—Teaching and Understanding Arithmetical Functions through Visualization

Szilárd Svitek <sup>1,\*</sup>, Norbert Annuš <sup>2,\*</sup> and Ferdinánd Filip <sup>1</sup> <sup>1</sup> Department of Mathematics, J. Selye University, 945 01 Komárno, Slovakia; filipf@ujs.sk<sup>2</sup> Department of Informatics, J. Selye University, 945 01 Komárno, Slovakia

\* Correspondence: svitek@ujs.sk (S.S.); annus@ujs.sk (N.A.)

**Abstract:** Number theory is an area of mathematics not unknown to students majoring in mathematics teaching. As early as in junior high, they may encounter basic number theory concepts such as prime numbers, multiples, divisors, or even the fundamental theorem of arithmetic—the proof of which, however, will only be included in the university curriculum of mathematics teacher students. When teaching number theory, in addition to inspiring and motivating students, it is important to have them ask the right questions. Proofs are important, but so is the proper formulation of mathematical conjectures, not to mention intuitions regarding mathematical properties and relationships. These processes are facilitated by the software used, developed to visualize functions related to number theory. Our study aims to assess the relevance of visualization in the teaching and understanding of certain arithmetic functions among mathematics teacher students. Furthermore, we aim to investigate the practical usability of our visualization software, relying on student feedback. According to our results, students found the software useful, as it helped them to discover previously unknown relationships in the examined arithmetic functions. We may conclude our software fills a niche, with student feedback indicating its usefulness.



**Citation:** Svitek, S.; Annuš, N.; Filip, F. Math Can Be Visual—Teaching and Understanding Arithmetical Functions through Visualization. *Mathematics* **2022**, *10*, 2656. <https://doi.org/10.3390/math10152656>

Academic Editors: Seifedine Kadry and Jay Jahangiri

Received: 21 June 2022

Accepted: 25 July 2022

Published: 28 July 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

**Keywords:** visualization; number theory; teacher training; formulation of conjecture; arithmetic function

**MSC:** 97F60; 97U50; 97D40

## 1. Introduction

We believe in the importance of creating as many situations as possible for students at all levels of education to experiment in the various areas of mathematics. When solving problems, students are willing to use experimentation or the trial and error approach, but typically with a certain lack of confidence. They usually hope to find solutions in the formulas or relationships they have already learned. As teachers, we need to emphasize that trial and error is a natural approach to solving mathematical problems and there is a lot to learn from it. Of course, “guiding” such adventures is a challenge as students are notorious for asking questions that teachers cannot answer right away.

This is particularly true in the case of number theory, where the solving of simple problems, easily understood even by students, may take centuries. A special advantage of problem-solving through trial and error is the use of digital tools at certain stages (their use might actually be essential due to computing demands). Students should gain as much experience as possible in working independently. Teachers should help students; however, this support should be neither too much nor too little, just enough to “get the students to the finish line” [1,2]. Computers, smartphones, and online presence are an integral part of the daily lives of students. We can utilize this by showing good examples of how to take advantage of computers and different software tools in mathematical problem-solving. For those who are also interested in programming, this type of task related to mathematical problems may be a particularly exciting challenge.

The number of divisors is addressed by numerous and diverse exercises as early as primary school. These exercises, to be solved in groups, are suitable even for students not specialized in mathematics, but simply interested in the subject. It is also important always to take the chance to explore mathematics in a playful way, using software tools wherever possible [3]. There are many opportunities to learn number theory through play, for example, playing games involving divisibility rules or number systems. We may create playful exercises to discover interesting number theory relationships, too. As regards the sum of divisors, perfect, amicable, and sociable numbers are sometimes mentioned as curiosities by textbooks, but the calculation of the sum of divisors using prime factorization is very rarely mentioned.

This playfulness and dynamism may also be transferred to university education. The lectures university students take and complete in mathematics are mostly introductory and are presented and discussed in a logical-deductive form. This often causes difficulties, especially for first-year students. Compared to secondary education, university studies represent a major change, as in the former system, the curriculum was rather practice-oriented, whereas in the latter case, the emphasis shifts to abstract aspects. It can, therefore, be useful and inspiring to introduce the new curriculum using practical exercises to engage the students.

As formulated by Pólya [2] (p. 105) “the solution often needs less insight and originality than the formulation”, while Dyson [4] (p. 163) described that in “the real world, most of the time, an answer is easier than defining a question”. We can therefore say that phrasing a good question is often more difficult than solving complex problems. According to Halmos [5,6], the most important task of teacher training is to explain to teach students how to teach their students in the future to formulate questions. This is, in fact, the basis of developing the attitude mathematics researchers need, i.e., students themselves should feel the urge to seek knowledge beyond the compulsory curriculum, driven by curiosity and a thirst for knowledge, to complement what they already know. A useful tool to achieve this is visualization. In this respect, a really useful software will provide more than just static images [7]. As in our research, similar activities using a spreadsheet were discussed by Abramovich and Brantlinger [8].

So the idea of using visualization to teach certain parts of mathematics is not new. One of the most widely used visualization software in education and research is GeoGebra, which can be used in disciplines such as geometry, algebra, and calculus at the university level [9]. Geogebra allows students to create their own illustrations for specific problems and explore them [10]. However, for example, Yamamoto et al. [11] investigated the potential of visualization for teaching recursive functions, Náhlík and Papoušková [12] used the advantages of visualization to teach parametrized functions in mathematics. The study of Rolfes et al. [13] confirms that visualizations in mathematics education can facilitate learning to some extent. Innovative approaches in the classroom are enhancing the potential of visualization in educational practice, thus giving it a more prominent role in mathematics education [14]. Using visualization to teach mathematics offers an exciting way of looking at mathematics in a way that has not been possible in the past [15].

The software used for our study [16] provides illustrations for a set of arithmetic functions, also supplemented with formulas and short descriptions—these latter ones were also used for the information material provided to the participants of the study. The functions included in the software are as follows.

## 2. Conceptual Framework

This section covers all the fundamentals of number theory that are relevant to the paper. Let us have the canonical form of  $n \in \mathbb{N}$  as  $n = p_1^{\alpha_1} \cdot p_2^{\alpha_2} \cdot \dots \cdot p_i^{\alpha_i} \cdot \dots \cdot p_k^{\alpha_k} > 1$ , where  $i, k \in \mathbb{N}$ . The arithmetic functions [17–19] to be used by us can be formulated as follows:

1. Divisor function—sum of the divisors  
In the case of the integer  $n$ , the sum of the positive divisors of the number  $n$  is given as  $\sigma(n)$ , where

$$\sigma(n) = \prod_{i=1}^k \left( \frac{p_i^{\alpha_i+1} - 1}{p_i - 1} \right).$$

2. Divisor function—the number of divisors

The number of divisors of the integer  $n$  is given as  $\tau(n)$  (or  $d(n)$  [17]), where

$$\tau(n) = \prod_{i=1}^k (\alpha_i + 1).$$

Secondary school students may find the symbol  $d(n)$  more familiar; this is why the software uses this one too. However, in the rest of this paper, we will use  $\tau(n)$ .

3. Divisor function—the product of the divisors

The product of all positive divisors of a nonzero integer  $n$  is equal to

$$P(n) = \prod_{d|n} d = n^{\frac{1}{2}\tau(n)},$$

where  $\tau(n)$  function expresses the number of the positive divisors of  $n$ .

4. Euler's totient function

Euler totient function  $\varphi(n)$  is an arithmetical function, which counts the number of positive integers smaller and coprime to  $n$ :

$$\varphi(n) = n \prod_{i=1}^k \left( 1 - \frac{1}{p_i} \right).$$

5. The Möbius function

For any positive integer  $n$ , we define the Möbius function as the sum of the primitive  $n^{\text{th}}$  roots of unity. We denote this function by  $\mu(n)$  and define it as follows:

$$\mu(n) = \begin{cases} 1 & \text{if } n=1, \\ 0 & \text{if } n \text{ has a square factor,} \\ (-1)^k & \text{if } n \text{ has } k \text{ distinct prime factors.} \end{cases}$$

6. Prime-counting function

We call  $\pi(n)$  the prime-counting function, which gives the number of prime numbers less than or equal to some integer  $n$ :

$$\pi(n) = \sum_{p_i \leq n} 1.$$

7. Distinct prime divisors

The  $\omega(n)$  function counts each distinct prime factor of  $n$ :

$$\omega(n) = \sum_{p|n} 1.$$

8. Liouville function

If  $n = p_1^{\alpha_1} \cdot p_2^{\alpha_2} \cdot \dots \cdot p_k^{\alpha_k}$ , then

$$\lambda(n) = (-1)^{\sum \alpha_i},$$

where  $\lambda(n)$  is the Liouville function and for  $\alpha_i$  applies  $1 \leq i \leq k$ . So its value is  $+1$  if  $n$  is the product of an even number of prime numbers, and  $-1$  if it is the product of an odd number of primes.

9. The prime gap counting function

The function  $g_n$  denotes the difference between two consecutive prime numbers i.e., the  $n$ th prime gap:

$$g_n = p_{n+1} - p_n.$$

The above functions have many beautiful and interesting properties. While it is not the purpose of this paper to describe these, we will refer to some later on to illustrate the possible uses of the software and to support the research with students. As already mentioned, these functions are not directly covered by the secondary curriculum, yet some textbooks contain sufficient knowledge so that students can understand them easily. The visualization of these functions promotes mathematics teaching in several ways:

- facilitates the recognition and understanding of the relationships between data,
- graphs and diagrams made from the data can be easily and accurately customized,
- compatible with Learning by Doing and modern teaching methods, such as problem-based and project-based learning [20].

### 3. Software Development

The software used for computation and visualization has been developed in the Lazarus programming environment. This is a free object-oriented programming environment, with an Object Pascal-based language of simple syntax. We chose this programming language because it is still used for programming exercises in Computer Science classes as well as at programming competitions. By using it, students can learn to work with objects and create simple applications quickly and easily. The software could therefore be used not only in teaching mathematics but as open source code, even in Computer Science classes, to perform simple simulation tasks, hence as a source of motivation or as a model application. Last but not least, as Computer Science classes as a school subject aims at developing algorithmic thinking and problem-solving skills, Lazarus could also be used for homework projects, for example, to write an algorithm with a similar purpose. The developed software, however, is designed to increase the efficiency of mathematics lessons, making the tasks simple and exciting. The importance and benefits of the digital visualization of the curriculum have been discussed above. However, it is also important to emphasize achieving the desired results using the software tool should be as simple as possible. Hereby we refer to the user-friendly interface of the software, which facilitates simple use. In designing our software, we have tried to make it expressly user-friendly, at the same time providing users with as much control as possible regarding the implementation of calculations and graphical representations. Such control options include the simple selection of functions, the option to define values in clearly labeled input fields, the possibility to zoom in on the desired interval in the graph when defining multiple values, as well as various graph types to select from. Such graph types include sequential simulations (with selectable speed) or ready-to-use graphs to instantly display the values of the chosen function within the entire tested interval. To illustrate all this from the users' point of view, the architecture of the software is shown in the Figure 1. The functions of individual objects are briefly described below.

Functions may be selected from a drop-down list (A), accessible via a ComboBox. Then the method for calculating and plotting the values may be selected. Options include sequential simulation, where  $i^{\text{th}}$  values in the function are displayed one at a time in a sequence and immediate calculations with ready-to-use graphs. CheckBox (B) serves the purpose of selecting from these options. When the sequential simulation is selected, the speed of the simulation may be controlled within a range of 100 and 1000 milliseconds using a Scrollbar (C). Actual values in the sequential simulation may be displayed using a Label (H). Once the general settings are selected, the values, i.e., the  $n$  parameters are defined via the input field Edit (E). Calculations and the simulations are run by pushing a Button (D). Parameters and return values are listed in a ListBox (F). The other objects are responsible for graphical display. A short description of the selected function and its mathematical treatment are displayed on a Canvas (G). Return values are graphically displayed on two Chart surfaces either as dots (J) or a line diagram (K). To handle intervals

easily, return values may also be listed by means of a CheckBox (I). This function is only available where return values are displayed as dots. This option may be activated or deactivated at any time, even before and after zooming in on an interval, or during sequential simulations.

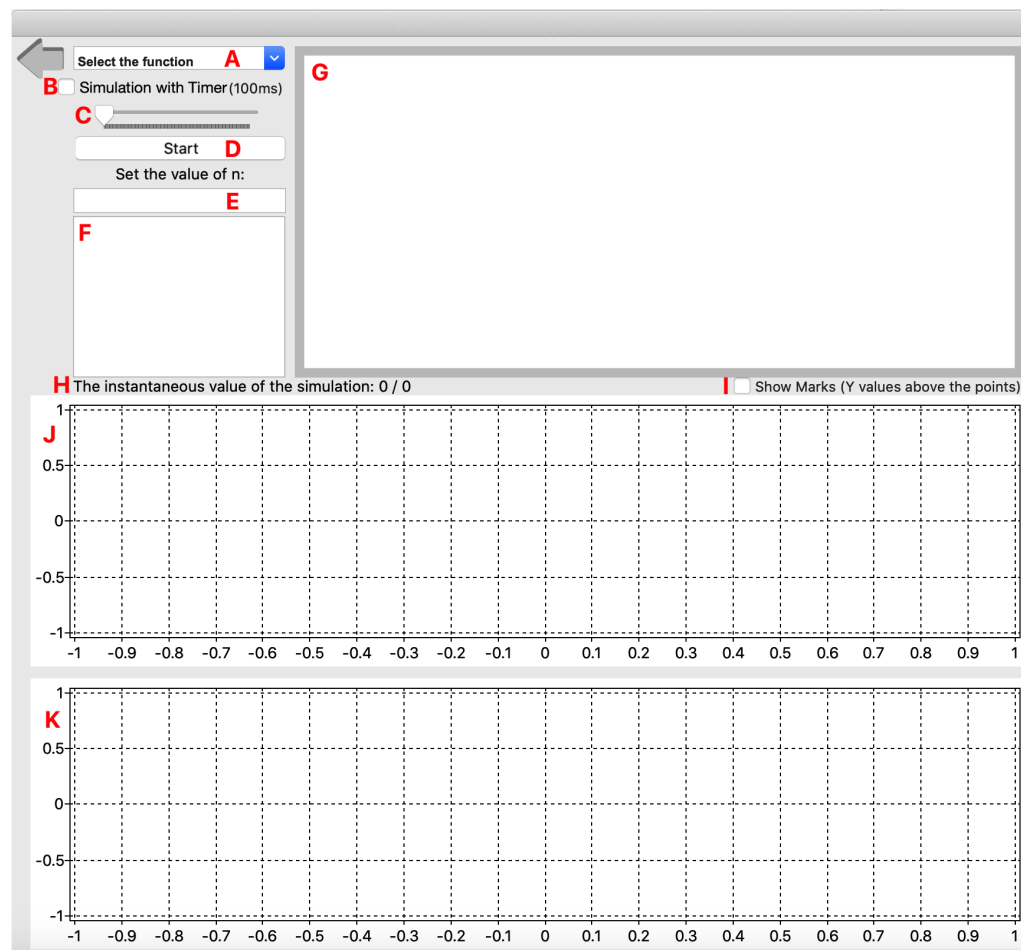


Figure 1. User interface of the software.

For the software, we have developed our own algorithms to calculate return values. In some cases it was easy to set up the calculation procedure. For example, for the functions on the divisors (number of divisors, sum of divisors, product of divisors), it was sufficient to implement the algorithm by checking the remainder of the division using a loop. If our condition that the remainder of the division (mod) is 0 is true, we incremented a value, summed or multiplied the number under test with the other values for which the condition was true, depending on the function. In other cases, e.g., for the Möbius and Euler functions, we had to rely on the relevant mathematical formulas and build the program code accordingly. From a functional point of view, the software is therefore capable of performing the following calculations as well as creating the related graphs (symbols used by the software are indicated in brackets): number of the divisors ( $\tau(n)$ ), sum of divisors and its variants ( $\sigma(n)$ ,  $\frac{\sigma(n)}{n}$ ), finding perfect numbers (during the tests, the calculation capacity was only sufficient up to  $n = 4$ ), Euler's totient function ( $\varphi(n)$ ), Möbius function ( $\mu(n)$ ), prime-counting function ( $\pi(n)$ ), product of divisors function ( $P(n)$ ), distinct prime divisors function ( $\omega(n)$ ), Liouville function ( $\lambda(n)$ ) and the prime gap counting function ( $g_n$ ). For each function, return values may be calculated and displayed for an interval between 1 and  $n$  at once. In the following sections, we demonstrate the operation of these functions as used by students for mathematical proofs.

#### 4. Software Implementation

Our study was conducted with students majoring in mathematics teaching ( $N = 7$ ). With them, group depth interviews were conducted online. These were recorded with the consent of the participants. The interviews were administered by the authors. The software was made available to the students two weeks before the survey. As they had only studied the functions  $\tau(n)$ ,  $\sigma(n)$ , and  $\varphi(n)$  beforehand, a short summary was provided for them on the other functions. This is the information material mentioned in the Introduction. The software was provided to the students with certain presets, but they were free to try other settings, too, regarding functions and intervals. We aimed at selecting simulation settings that could reveal some correlation or property about the number theory functions to motivate them to explore the software. This setup was intended to imitate the role of teachers, as the participating students were not assisted by anybody while exploring the properties of the software.

The mathematical conjectures formulated by the students from the simulations will be discussed as follows:

- the settings for the visualization are provided and illustrated by figures,
- the corresponding insights and conjectures by the student are provided,
- these conjectures are confirmed/refuted (where possible),
- any further comments are provided, if there were any.

Simulations are presented here in the same order as they were encountered by the students. This is particularly important because some conjectures were formulated relying on ideas inspired by some earlier simulations. Students were allowed to work both individually or in groups which resulted in some conjectures pointing to the same thing, but approached from different angles. This may be an indication of the student's way of thinking, so it is worth presenting them separately. This also shows how the stimulus provided by visualization may differ for each student. Some proofs may seem trivial, but the verification of the conjecture is as important as the formulation of the conjecture itself. Students reported that the positive feedback received when developing the proofs motivated them to work even harder. Even when a conjecture was refuted, it was considered by them as motivation rather than discouragement.

##### 4.1. Simulation 1

Based on Figure 2, students made the following conjectures.

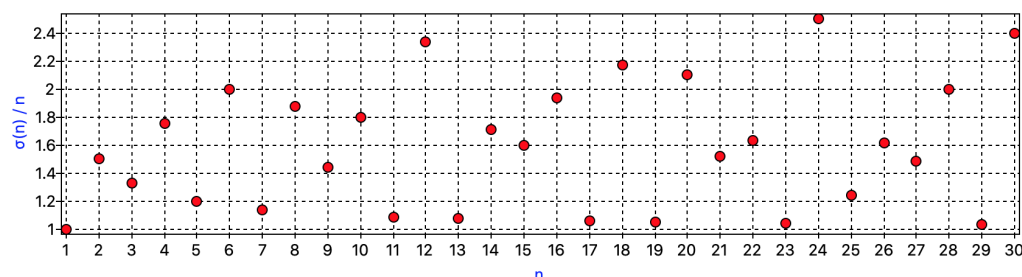
**Conjecture 1.** *In the case of prime numbers, this ratio decreases.*

**Conjecture 2.** *For prime numbers, this value converges to one.*

**Proof.** We know, that if  $n$  is a prime number  $p$ , then  $\sigma(p) = p + 1$ . Then  $\lim_{p \rightarrow \infty} \frac{p+1}{p} = 1$ .  $\square$

**Remark 1.** *The Conjectures 1 and 2 are rather similar. The second one is more accurate, including not only the decrease of the ratio  $\frac{\sigma(n)}{n}$  in the case of prime numbers, but also that this sequence is bounded from below.*

**Remark 2.** *Conjecture 2 applies to any  $n \in \mathbb{N}$  if there is a lower bound, as for all  $n > 2$  holds, that  $1 < \frac{\sigma(n)}{n}$ . Then  $\liminf_{n \rightarrow \infty} \frac{\sigma(n)}{n} = 1$ .*



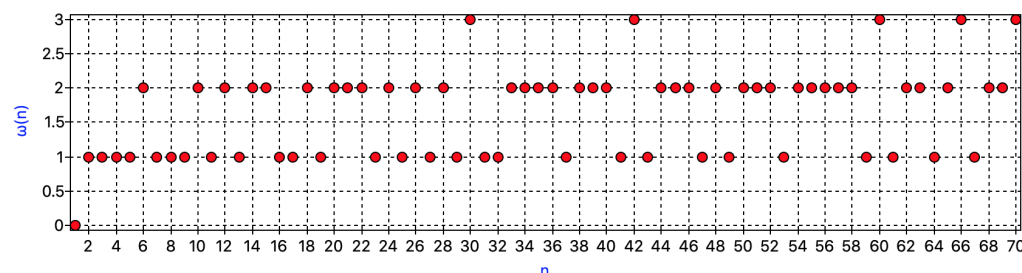
**Figure 2.** The simulation for  $\frac{\sigma(n)}{n}$  with  $1 \leq n \leq 30$ .

#### 4.2. Simulation 2

Based on Figure 3, students made the following conjecture.

**Conjecture 3.** All values from 2 onwards will fall between  $\omega(n) = 1$  and  $\omega(n) = 3$ .

**Remark 3.** The statement is true for the specified interval, however, when testing it for a broader one (e.g.,  $n = 1000$ ), counterexamples are readily found. Accordingly, the statement may be refuted easily, as for  $n = 210$  when  $\omega(n) = 4$ . Anyhow, this is a good example of how the use of the software motivated students to find new properties and relationships, even if some of these were not, in fact, “new”.



**Figure 3.** The simulation for  $\omega(n)$  with  $1 \leq n \leq 70$ .

#### 4.3. Simulation 3

Based on Figure 4, students made the following conjectures.

**Conjecture 4.** For the powers of 2,  $\sigma(n)$  is always an odd number.

**Proof.** We know that the divisors of numbers given as  $2^k$  are even numbers (except for 1):  $2^1, 2^2, 2^3, \dots, 2^k$ . Accordingly  $\sigma(2^k) = 1 + 2 + 4 + \dots + 2^k$ , which sum will always be odd.  $\square$

**Remark 4.** The proof is trivial, yet students will find it motivating to see their conjecture was right (even if this conjecture is not something revolutionary).

**Conjecture 5.** The sum of divisors for square numbers are always an odd number.

**Proof.** Let  $n^2 = p_1^{2\alpha_1} \cdot p_2^{2\alpha_2} \cdot \dots \cdot p_i^{2\alpha_i} \cdot \dots \cdot p_k^{2\alpha_k}$ . In this case

$$\sigma(n^2) = (p_1^0 + p_1^1 + \dots + p_1^{2\alpha_1})(p_2^0 + p_2^1 + \dots + p_2^{2\alpha_2}) \dots (p_k^0 + p_k^1 + \dots + p_k^{2\alpha_k}).$$

Note that the parentheses contain the sum of an even number of prime powers and 1, which is always an odd sum. It means odd numbers are multiplied, the product of which is always an odd number.  $\square$

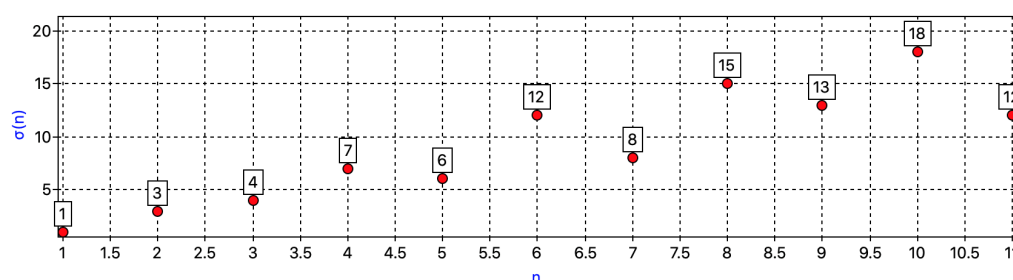
**Remark 5.** Regarding this conjecture, the students put forward a weaker statement, too: If  $n$  is a prime number, then  $\sigma(n^2)$  is an odd number. This is a special case of the previous statement, so it will not be discussed separately.

**Conjecture 6.** In the case of even numbers  $\sigma(n)$  is always greater than that of the even number before it.

**Remark 6.** This statement is true in the specific interval, yet counterexamples are readily found when extending it:  $\sigma(12) = 28$  and  $\sigma(14) = 24$ . It is always worth examining such conjectures because they illustrate them, which is not enough to stop at putting forward the conjecture; instead, it should be proven or refuted by counterexamples, using the available means—be it some software tool or our existing knowledge.

**Conjecture 7.** In the case of prime numbers,  $\sigma(n)$  is greater than the prime itself by 1.

**Remark 7.** The  $\sigma(p) = p + 1$  is a known property of the  $\sigma(n)$  function in the case of prime numbers. The definition of the function is, in fact, the proof of this statement. Although a trivial proof, finding it is highly motivating for students, as in the case discussed under Conjecture 10, the student referred to this conjecture as a source of inspiration.



**Figure 4.** The simulation for  $\sigma(n)$  with  $1 \leq n \leq 11$ .

#### 4.4. Simulation 4

Based on Figure 5, students made the following conjectures.

**Conjecture 8.** The return value for the doubles of prime numbers is always 1.

**Proof.** The  $\mu(n)$  function equals to 1, when  $k$  is an even number in the case of  $(-1)^k$ , i.e., there are no square numbers in  $n$  and the number of prime factors is even. A prime number has exactly one prime divisor; accordingly, for prime numbers  $\mu(n) = -1$ . As the prime number in question is multiplied by the only even prime, it will have two prime factors, i.e.,  $k = 2$ . Then  $\mu(2p) = (-1)^2 = 1$ . This is not true for  $p = 2$ , as  $2 \times 2 = 4$ , which is a square number, so  $\mu(4) = 0$ .  $\square$

**Remark 8.** The statement by the student was false for  $p = 2$ , it will be true nevertheless when the precondition  $p > 2$  is applied, as illustrated above.

**Conjecture 9.** For the multiples of 4, the value for  $\mu(n)$  is always 0.

**Remark 9.** The statement is trivial as 4 will be a divisor of the natural number  $n$ , which by definition takes  $\mu(n) = 0$ . The statement would actually be true for any square number.

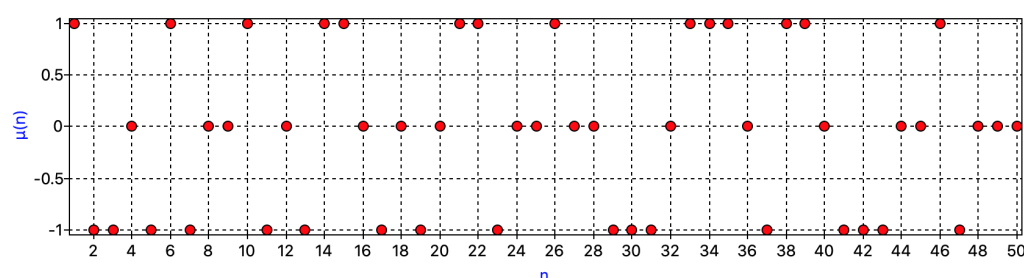


Figure 5. The simulation for  $\mu(n)$  with  $1 \leq n \leq 50$ .

#### 4.5. Simulation 5

Based on Figure 6, students made the following conjectures.

**Conjecture 10.** In the case of prime numbers, the value of  $\varphi(n)$  will be less than the given prime number by 1.

**Remark 10.** It is relatively easy to confirm this statement as a  $p$  integer will be a prime when each lesser positive integer is a relative prime; otherwise, it would have a prime divisor less than it. Accordingly, the known formula of  $\varphi(n) \leq n - 1$  is phrased as  $\varphi(p) = p - 1$  for prime numbers.

**Conjecture 11.** The value of the function  $\varphi(n)$  increases between two consecutive integers if the larger one is a prime number.

**Remark 11.** This conjecture by the student is quantified as  $\varphi(p - 1) < \varphi(p)$ . The statement may be confirmed by Remark 10, where the known relationship  $\varphi(p) = p - 1$  was stated for prime numbers. In that case,  $\varphi(p - 1) < p - 1$ . This inequation is always true when  $p > 3$ . When  $p = 3$ , the original inequation is transformed as follows (with 2 and 3 being the only consecutive prime numbers):  $\varphi(2) < \varphi(3)$ , and this is of course true.

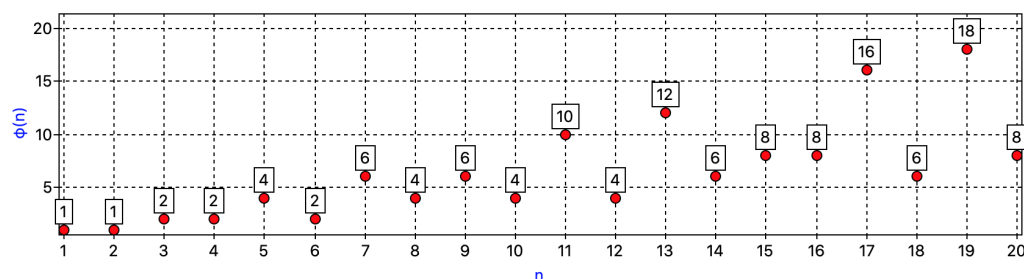


Figure 6. The simulation for  $\varphi(n)$  with  $1 \leq n \leq 20$ .

#### 4.6. Simulation 6

Based on Figure 7, students made the following conjecture.

**Conjecture 12.** In the case of the Möbius function, the values of the function will be equal for maximum of 3 consecutive natural numbers.

**Remark 12.** The conjecture is easily confirmed for  $\mu(n) = -1$  and  $\mu(n) = 1$ , since there will be one number among four consecutive ones which is divisible by 4, so the value of the function will be  $\mu(n) = 0$  for at least one of them. However, for  $\mu(n) = 0$  the conjecture is no longer valid. Using the Chinese remainder theorem [21], it is easily proven that  $k$  consecutive natural numbers may be defined, where

$$\mu(n + 1) = \mu(n + 2) = \mu(n + 3) = \dots = \mu(n + k) = 0.$$

For example, in the case of  $n = 29,347$ , there are five consecutive  $\mu(n) = 0$  values:  $\mu(29,348) = \mu(29,349) = \mu(29,350) = \mu(29,351) = \mu(29,352) = 0$  (see Figure 8).

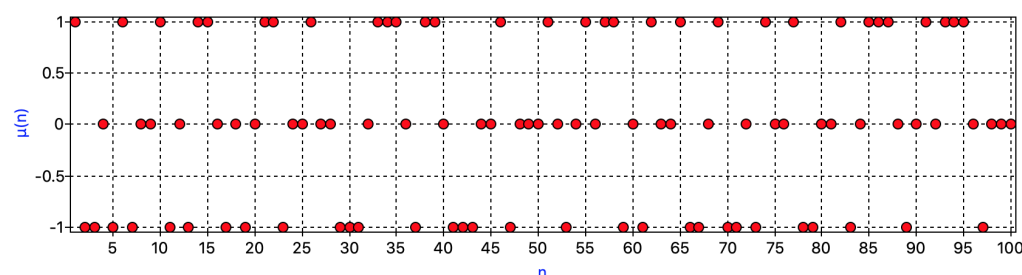


Figure 7. The simulation for  $\mu(n)$  with  $1 \leq n \leq 100$ .

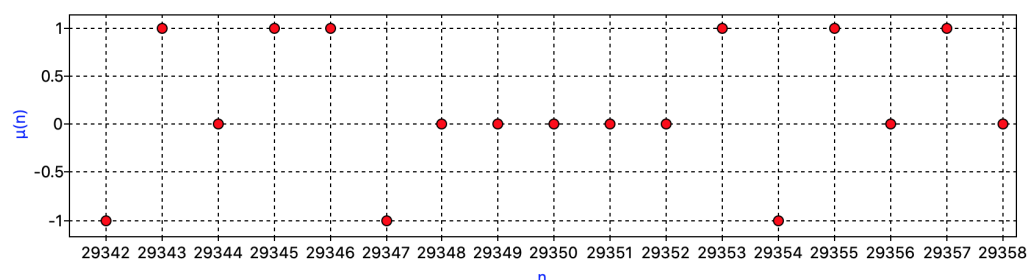


Figure 8. The simulation for  $\mu(n)$  and  $1 \leq n \leq 29,360$ , zooming in on the value  $n = 29,350$ .

#### 4.7. Simulation 7

Based on Figure 9, students made the following conjectures.

**Conjecture 13.** The value  $-1$  is encountered more frequently than 1.

**Remark 13.** This conjecture was also formulated by George Pólya [22] as follows:

Let  $L(n) = \sum_{k=1}^n \lambda(k)$ . Then  $L(n) \leq 0$  for each  $n > 1$ .

This was disproved by Tanaka [23], who provided the lowest counterexample  $n = 906,150,257$ .

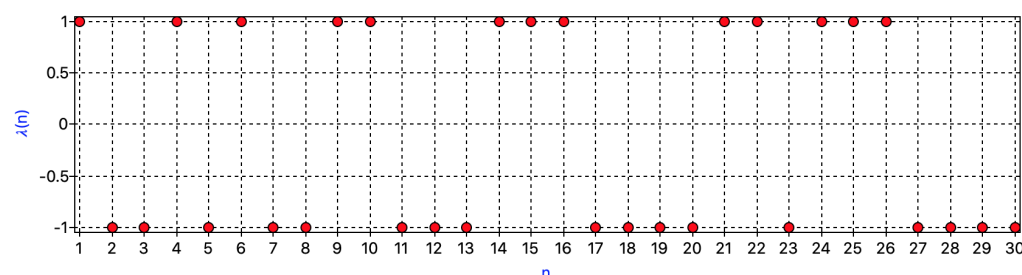
**Conjecture 14.** For square numbers, the value of the function is always 1.

**Proof.** Let us have  $n = p_1^{\alpha_1} \cdot p_2^{\alpha_2} \cdot \dots \cdot p_k^{\alpha_k}$ . In that case  $n^2 = p_1^{2\alpha_1} \cdot p_2^{2\alpha_2} \cdot \dots \cdot p_k^{2\alpha_k}$ , therefore  $2\alpha_1 + 2\alpha_2 + \dots + 2\alpha_k$  would be the exact quantity of the prime numbers, which we multiply. Then we can obtain  $2(\alpha_1 + \alpha_2 + \dots + \alpha_k)$ . Obviously, the exponent is an even number, so  $\lambda(n) = 1$ . We should also note here that this result does not depend on the exponent of the prime factors in this case.  $\square$

**Conjecture 15.** In the case of cube numbers and prime numbers, the value of the function is always  $-1$ .

**Remark 14.** This conjecture is not true for cube numbers. Counterexamples may be readily found, e.g.,  $n = 64$ , where the integer is factorized as  $64 = 2^6$ , so  $\lambda(n) = 1$ .

**Remark 15.** The conjecture is obviously true for prime numbers as a prime number is “made up” of exactly one prime number.



**Figure 9.** The simulation for  $\lambda(n)$  with  $1 \leq n \leq 30$ .

#### 4.8. Simulation 8

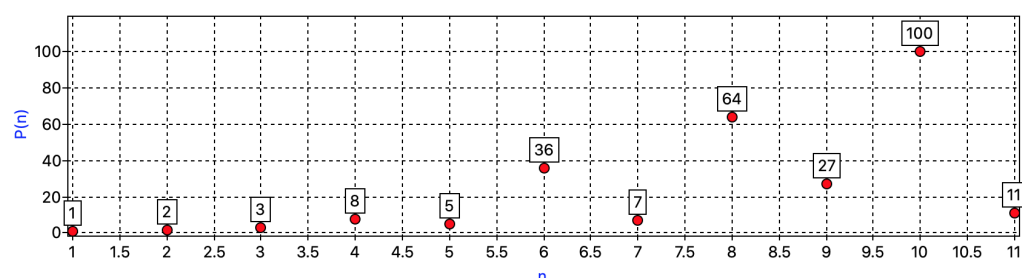
Based on Figure 10, students made the following conjectures.

**Conjecture 16.** *The value of the  $P(n)$  function increases for prime numbers.*

**Remark 16.** *This conjecture follows from the definition, as the  $P(n)$  function is defined as the product of the positive divisors of an integer, i.e.,  $P(p) = p$ , where  $p$  is a prime number.*

**Conjecture 17.** *The value of the  $P(n)$  function increases for even numbers.*

**Remark 17.** *This conjecture can not be true, as counterexamples may be readily found:  $P(12) = 1728$  and  $P(14) = 196$ ; hence there is no monotonic increase. When working with the students, it was important to emphasize that results obtained with small samples must not be regarded as conclusive. This conjecture was a good example of that.*



**Figure 10.** The simulation for  $P(n)$  with  $1 \leq n \leq 11$ .

#### 4.9. Simulation 9

Based on Figure 11, students made the following conjectures.

**Conjecture 18.** *For  $n > 1$ , the value of the  $\tau(n)$  function will be even, except for square numbers.*

**Proof.** Let us have the following statement, considered identical to the conjecture to be proven:  $\tau(n)$  is odd only then when  $n$  is a square number. Let  $n^2 = p_1^{2\alpha_1} \cdot p_2^{2\alpha_2} \cdot \dots \cdot p_k^{2\alpha_k}$ . Then, according to the definition of  $\tau(n)$

$$\tau(n) = (2\alpha_1 + 1)(2\alpha_2 + 1) \dots (2\alpha_k + 1).$$

Note that each multiplicand is odd, i.e., their product may only be an odd number.  $\square$

**Remark 18.** *Based on the previous conjecture, another statement was made by the students: square numbers always yield odd return values, which is true.*

**Conjecture 19.** *For prime numbers  $p > 2$  multiplied by two, the value of the  $\tau(n)$  function is 4.*

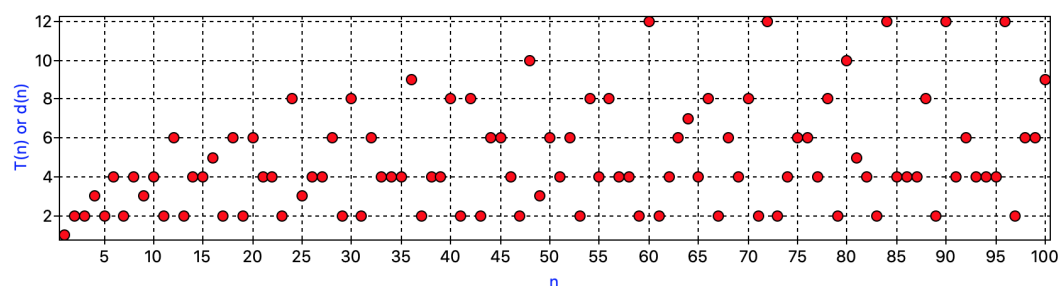
**Remark 19.** This conjecture can be easily proved. Let us phrase it as  $\tau(2p) = 4$ , where  $p > 2$  and is a prime number. Utilizing the multiplicative property of the  $\tau(n)$  function, this expression may be written out as

$$\tau(2p) = \tau(2^1) \cdot \tau(p^1) = (1+1)(1+1) = 4.$$

In connection with this conjecture, the students made another one: the value of the function for prime numbers will always be 2. We will not discuss this conjecture separately as the proof is obviously similar to the previous one, and the two statements are closely related.

**Conjecture 20.** The only two consecutive integers where  $\tau(n) = 2$  is true are  $n = 2$  and  $n = 3$ .

**Remark 20.** The condition  $\tau(n) = 2$  implies that  $n$  is a prime number, as the only numbers with exactly two divisors are the primes. Accepting this, the conjecture is easily confirmed as the only two consecutive primes are 2 and 3.



**Figure 11.** The simulation for  $\tau(n)$  with  $1 \leq n \leq 100$ .

#### 4.10. Student Feedback

- “The software would be a useful tool for number theory classes, rendering functions a bit easier to grasp.”
- “The software makes the functions in number theory more comprehensible.”
- “The theorems and properties we are studying became more tangible. By actually seeing them, we can understand them better.”
- “It would be good if students were given a chance to figure out the theorems before they are stated.”
- “Visualization helped a lot, it is much easier than just reading.”
- “It’s similar to Geogebra in that you can use it not only to calculate, but also to see where the values are located in relation to each other.”

### 5. Discussion

The visualization software can make number theory functions more tangible for students, giving them a better grasp of the relationships regarding the return values of various functions. The software may be used to formulate conjectures and notice properties and patterns. Accordingly, it does not limit the user to recognizing known relationships only but also offers the possibility of discovering something new. A further benefit of the software visualizing number theory functions is that students can easily check their results after completing a homework assignment if they are not sure of themselves or just want to double-check. In case of a mismatch between their own results and those given by the software, students have the possibility to track where they failed, knowing the correct result.

We plan to introduce the visualization software to high school students in the future, as the secondary curriculum includes several of the functions we discussed here. Thus, we tried to use as simple tools as possible when confirming statements so that the proofs can be used in our further research with high school students. After all, they are unlikely to have any serious mathematical apparatus yet.

As we can see, students provided with some direction and properly formulated insights may come up with anything, from the most trivial statements to complex properties. By formulating simpler statements, the knowledge to be acquired may be captured more efficiently, whereas confirming more sophisticated theorems, definitions, or properties does not only build on the knowledge acquired in the previous steps, but also allows for a sneak peek of the vistas of mathematics.

As the results show, the students were able to formulate their conjectures easily [4], although not always accurately, but this may be due to a lack of routine. As expected, less background knowledge was sufficient to formulate the conjectures [2]. Furthermore, visualization clearly has a place in mathematics education, offering innovative classroom approaches [14] and exciting new ways of visualizing [15], which can show the true delights of mathematics. The only question is in what proportion and where it can be used most effectively.

Based on our current results, we will also integrate the software into the calculus course according to our possibilities. We also plan to involve more universities to reach a larger number of students. We need to carry out a complete educational intervention using the developed tool to obtain verifiable results in terms of its didactic efficacy.

**Author Contributions:** Conceptualization, S.S. and N.A.; Funding acquisition, S.S.; Investigation, S.S. and N.A.; Methodology, S.S. and N.A.; Software, N.A.; Supervision, F.F.; Visualization, N.A.; Writing—original draft, S.S. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by VEGA grant (Vedecká Grantová Agentúra MŠVVaŠ SR) number 1/0386/21 and was supported by the Pallas Athéné Domus Meriti Foundation.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Informed consent was obtained from all subjects involved in the study.

**Data Availability Statement:** Not applicable.

**Acknowledgments:** We would like to thank the anonymous reviewers who took the time and effort to review the manuscript. We sincerely appreciate all their valuable comments and suggestions that helped to improve the quality of the manuscript.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Pólya, G. *How to Solve It*; Princeton University Press: Princeton, NJ, USA, 1945.
2. Pólya, G. *Mathematical Discovery: On Understanding, Learning and Teaching Problem Solving*, Combined ed.; Wiley: New York, NY, USA, 1981.
3. Csiba, P. O vyučovaní matematiky na prelome tretieho tisícročia. In Proceedings of the Zborník z I. Medzinárodnej Vedeckej Konferencie Univerzity J. Selyeho “Vzdelávanie-veda-Spoločnosť”, Komárno, Slovakia, 7–8 September 2009; pp. 38–42.
4. Dyson, G. *Turing’s Cathedral: The Origins of the Digital Universe*; Pantheon Books: New York, NY, USA, 2012.
5. Halmos, P.R. The teaching of problem solving. *Am. Math. Mon.* **1975**, *82*, 466–470. [\[CrossRef\]](#)
6. Halmos, P.R. Jádno matematiky. *Pokrok. Mat. Fyziky A Astron.* **1982**, *27*, 273–280.
7. Borwein, P.; Jörgenson, L. Visible Structures in Number Theory. *Am. Math. Mon.* **2001**, *108*, 897–910. [\[CrossRef\]](#)
8. Abramovich, S.; Brantlinger, A. Technology-motivated teaching of topics in number theory through a tool kit approach. *Int. J. Math. Educ. Sci. Technol.* **2004**, *35*, 317–333. [\[CrossRef\]](#)
9. Ziatdinov, R.; Valles, J.R., Jr. Synthesis of Modeling, Visualization, and Programming in GeoGebra as an Effective Approach for Teaching and Learning STEM Topics. *Mathematics* **2022**, *10*, 398. [\[CrossRef\]](#)
10. Gunčaga, J.; Zawadowski, W.; Prodromou, T. Visualisation of Selected Mathematics Concepts with Computers—The Case of Torricelli’s Method and Statistics. *Eur. J. Contemp. Educ.* **2019**, *8*, 69–91.
11. Yamamoto, R.; Anzai, Y.; Kogure, S.; Noguchi, Y.; Yamashita, K.; Konishi, T.; Itoh, Y. Learning Environment for Recursive Functions by Visualization of Execution Process. In Proceedings of the 25th International Conference on Computers in Education, Christchurch, New Zealand, 4–8 December 2017; pp. 421–426.
12. Náhlík, T.; Papoušková, K. Visualization of Parametric Functions as a Tool for Teaching Mathematics. In Proceedings of the 10th International Conference on Education and New Learning Technologies, EDULEARN18 Proceedings, Palma, Spain, 2–4 July 2018; pp. 6289–6294.

13. Rolfes, T.; Roth, J.; Schnotz, W. Learning the Concept of Function with Dynamic Visualizations. *Front. Psychol.* **2020**, *11*, 693. [[CrossRef](#)]
14. Arcavi, A. The role of visual representations in the learning of mathematics. *Educ. Stud. Math.* **2003**, *52*, 215–241. [[CrossRef](#)]
15. Epperson, J.; Byrns, A. Studying The Role of Dynamic Visualization in Learning Transformations of Functions. In Proceedings of the 6th International Conference on Education and New Learning Technologies, EDULEARN14 Proceedings, Barcelona, Spain, 7–9 July 2014; pp. 390–399.
16. Annuš, N.; Svitek, S. Visualization of Arithmetical Functions in Education. In Proceedings of the 16th International Technology, Education and Development Conference, INTED2022 Proceedings, Valencia, Spain, 7–8 March 2022; pp. 2047–2053.
17. Hardy, G.H.; Wright, E.M. *An Introduction to the Theory of Numbers*, 5th ed.; Oxford University Press: Oxford, UK, 1979.
18. Apostol, T.M. *Introduction to Analytic Number Theory*; Springer: New York, NY, USA, 1976.
19. László, B.; Tóth, J. *Bevezetés a Szárméletbe*; Lilium Aurum: Dunajská Streda, Slovakia, 1999.
20. Csóka, M. Data Visualization as Part of High School Programming. In Proceedings of the 13th International Conference on Education and New Learning Technologies, EDULEARN21 Proceedings, Online, 5–6 July 2021; pp. 9484–9489.
21. Ireland, K.; Rosen, M. The Chinese Remainder Theorem. In *A Classical Introduction to Modern Number Theory*, 2nd ed.; Springer: New York, NY, USA, 1990; pp. 34–38.
22. Pólya, G. Verschiedene Bemerkungen zur Zahlentheorie. *Jahresber. Dtsch. Math.-Ver.* **1919**, *28*, 31–40.
23. Tanaka, M. A Numerical Investigation on Cumulative Sum of the Liouville Function. *Tokyo J. Math.* **1980**, *3*, 187–189. [[CrossRef](#)]