

Fibonacci Numbers and Selected Practical Applications in the Matlab Computing Environment

Viliam Ďuriš^{a*} – Lucia Rumanová^b – Dusan Vallo^c – Júlia Záhorská^d

^{a,b,c,d,*} Department of Mathematics, Faculty of Natural Sciences, Constantine the Philosopher University in Nitra, Tr. A. Hlinku 1, 949 74 Nitra, Slovakia

Received March 14, 2019; received in revised form March 25, 2019; accepted April 4, 2019

Abstract

Fibonacci numbers play an important role in mathematics and appear in many places to solve practical problems even. In fact, many complex publications have been published on this issue. Surprisingly, Fibonacci numbers are used so extensively that an association dealing with issues related to them was established. Since 1963, it has issued a specialized scientific journal called Fibonacci Quarterly. The most well-known applications that take advantage of Fibonacci numbers are in geometry, but when exploring their contexts and methods, the use of Fibonacci numbers in computer science and in school mathematics has also been addressed.

Keywords: Fibonacci, Matlab, golden ratio, encryption, unit transfers

Classification: I20, C60

Introduction

The Italian mathematician Leonardo da Pisa (also known as Fibonacci), who lived in the early 13th century, became famous in the history of mathematics for unriddling the following task: *“Someone has placed a couple of rabbits in sort of an enclosure to find out how many couples are born within a year when the couple of rabbits brings another couple into the world on a monthly basis, considering the rabbits begin to give birth when they are two months old”* (Knott, Quinney, 1997). Fibonacci solved this problem as follows. In the first and second month, there is only one pair of rabbits. In the third month, it is two couples, because the original couple brings a new couple (a baby boy rabbit and a baby girl rabbit) after two months. In the fourth month, there are three couples because it is only the original couple that still bears offspring. In the fifth month, we have five couples, because two more couples are added to those that lived in the fourth month, because in the fifth month the couples that lived in the third month will give birth. In the sixth month, except for the couples that already existed in the fifth month, three more couples will come in, because the fourth-month couples will give birth. And that is how it goes (Figure 1) (Fulier, Šedivý, 2001).

Fibonacci realized that the number of couples in a given month was the sum of the number of couples that lived in the previous month and the number of couples that already lived two months earlier. He marked the number of couples in the n -th month as a_n . Then the rule he found could be written as $a_n = a_{n-1} + a_{n-2}$, with only $n > 2$ allowed under this rule.

*Corresponding author; email: vduris@ukf.sk
DOI: 10.17846/AMN.2019.5.1.14-22

Fibonacci then easily calculated $a_7 = a_6 + a_5 = 8 + 5 = 13$, $a_8 = a_7 + a_6 = 13 + 8 = 21$, $a_9 = a_8 + a_7 = 21 + 13 = 34$, $a_{10} = a_9 + a_8 = 34 + 21 = 55$, $a_{11} = a_{10} + a_9 = 55 + 34 = 89$ and eventually a element $a_{12} = 144$, and thus the answer that there would be 144 rabbits after a year.

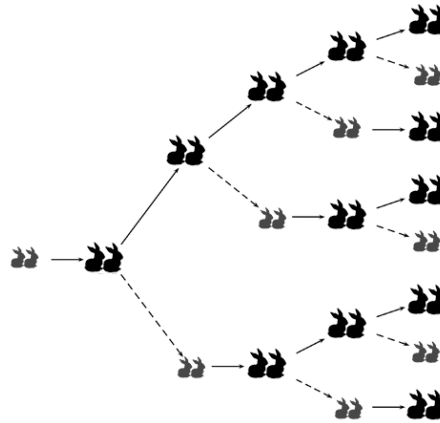


Fig. 1: Fibonacci's rabbit task

2 Fibonacci numbers and the golden ratio in geometry

Let us look at the sequence

$$1, 1, 2, 3, 5, 8, \dots$$

where for every element it is possible to state:

$$a_n = a_{n-1} + a_{n-2}, n > 2, a_1 = a_2 = 1$$

The Fibonacci sequence could easily be generated, for instance, in the Matlab computing environment with the following algorithm (Dikovic, 2017):

```
function f = fibonacci(n)
    f = zeros(n, 1);
    f(1) = 1;
    f(2) = 1;
    for k = 3:n
        f(k) = f(k - 1) + f(k - 2);
    end
end
```

The Fibonacci function in Matlab can also be called recursively. However, a recursive call prolongs the calculation and often blocks a large memory area by reserving a memory location to create temporary local variables.

```
function f = fibonacci2(n)
    if n == 1 | n == 2
        f = 1;
        return
    end
    f = fibonacci2(n - 1) + fibonacci2(n - 2);
end
```

Between the Fibonacci sequence and the geometric sequence

$$b, bq, bq^2, bq^3, \dots$$

there exists an analogy (Znám, 1975), because for the first three elements of the geometric sequence there can be applied that

$$b + bq = bq^2, \text{ hence } 1 + q = q^2, b \neq 0$$

Then each element of the sequence bq^2, bq^3, \dots is from the sum of the previous two and there applies

$$bq^{n-2} + bq^{n-1} = bq^n, n \geq 2$$

Our task is to properly choose a number q to get elements of the Fibonacci sequence.

Let us say that the sequence $\{Z_u\}_{u=1}^{\infty}$ has a property I if $u \geq 3$, it holds true that

$$Z_u = Z_{u-1} + Z_{u-2}.$$

If then the following sequences

$$x_1, x_2, x_3, \dots$$

$$y_1, y_2, y_3, \dots$$

have the property I , so does the sequence

$$x_1 + y_1, x_2 + y_2, x_3 + y_3, \dots$$

have the property I . Then, the sequence $\{cq_1^n + dq_2^n\}$ also has the property I . Therefore, the numbers c and d have to be chosen the way that the first two elements of this sequence equal the numbers a_1 and a_2 , hence the following equalities are valid

$$cq_1 + dq_2 = a_1 = 1$$

$$cq_1^2 + dq_2^2 = a_2 = 1$$

From $b + bq = bq^2$ it is valid:

$$q_1 = \frac{1 + \sqrt{5}}{2}, q_2 = \frac{1 - \sqrt{5}}{2}$$

After rearrangement we get:

$$c(1 + \sqrt{5}) + d(1 - \sqrt{5}) = 2$$

$$c(6 + 2\sqrt{5}) + d(6 - 2\sqrt{5}) = 4$$

By solving this system of equations we get:

$$c = \frac{1}{\sqrt{5}}, d = -\frac{1}{\sqrt{5}}$$

We got a straightforward number a_n :

$$a_n = \frac{1}{\sqrt{5}} \left\{ \left(\frac{1 + \sqrt{5}}{2} \right)^n - \left(\frac{1 - \sqrt{5}}{2} \right)^n \right\}$$

Then for each natural number n it is valid

$$a_1 + a_2 + \dots + a_n = a_{n+2} - 1$$

If we denote $A = \frac{1 + \sqrt{5}}{2}, B = \frac{1 - \sqrt{5}}{2}$, using the relation for the expression a_n and the fact that $\frac{1}{A} = -B, A = 1 - B$ we get

$$\begin{aligned} a_1 + \dots + a_n &= \frac{1}{\sqrt{5}} (A + A^2 + \dots + A^n) - \frac{1}{\sqrt{5}} (B + B^2 + \dots + B^n) = \\ &= \frac{1}{\sqrt{5}} \frac{A^{n+1} - 1}{A - 1} - \frac{1}{\sqrt{5}} \frac{B^{n+1} - 1}{B - 1} = \frac{1}{\sqrt{5}} \left\{ \frac{A^{n+1}}{-B} - \frac{B^{n+1}}{-A} + \frac{1}{B} - \frac{1}{A} \right\} = \\ &= \frac{1}{\sqrt{5}} \{A^{n+2} - B^{n+2}\} - 1 = a_{n+2} - 1 \end{aligned}$$

An important finding of the Fibonacci numbers is that two neighbouring Fibonacci numbers are always relatively prime. We are able to prove it using mathematical induction. For the first

two Fibonacci numbers, it is valid that they are relatively prime. Let us suppose now that a_{n-1} and a_n are relatively prime. Then, from the equality $a_{n+1} = a_n + a_{n-1}$ there follows that every common divisor of the numbers a_n and a_{n+1} would also have to be a divisor for the number a_{n-1} , and therefore also a common divisor of the numbers a_n and a_{n-1} . Hence, $(a_n, a_{n+1}) = 1$.

Additionally, it is valid that the number a_n is a divisor of the number a_m only and when n is a divisor of the number m . We can first prove that if $n|m$, then $a_n|a_m$. Let us consider $n|m$ and $m = nm_1$. Let us proceed by induction with respect to m_1 . If $m_1 = 1$, then $m = n$ and the theorem is valid. Let us suppose that our theorem is valid for $m_1 = k$. We can prove that it also applies to $m_1 = k + 1$. If u and v are natural numbers, then $a_{u+v} = a_{u-1}a_v + a_u a_{v+1}$. By using this relation and by substituting $u = mk, v = m$ we get:

$$a_{m(k+1)} = a_{mk+m} = a_{mk-1}a_m + a_{mk}a_{m+1}$$

On the right hand side, both of the addends are divisible by the number a_m based on the inductive assumption $a_m|a_{m(k+1)}$. Now, we can prove that if $a_n|a_m$, then $n|m$. Let then $a_n|a_m$ and let $m = n \cdot s + t, 0 < t < n$. By substituting $u = ns, v = t$ we get:

$$a_m = a_{ns-1}a_t + a_{ns}a_{t+1}$$

From this equality, it follows that $a_n|a_{ns-1}a_t$. According to the proven part of the theorem, it is valid that $a_n|a_{ns}$ and according to the assumption of the theorem $a_n|a_m$. Furthermore, we know that $(a_{ns}, a_{ns-1}) = 1$ is valid, therefore $a_n \nmid a_{ns-1}$ (because it is a divisor of the number a_{ns}). Then $a_n|a_t$. However, this is not possible because the Fibonacci numbers form a growing sequence (from $n = 2$), and therefore $a_n > a_t$, because $n > t$.

The findings about Fibonacci numbers allow us to solve problems related to the golden ratio in geometry. We say that the sides of the rectangle a, b form the golden ratio if

$$\frac{a+b}{a} = \frac{a}{b}$$

is applied. After adjusting the last equality, we get $a^2 = ab + b^2$ or

$$\left(\frac{a}{b}\right)^2 - \frac{a}{b} - 1 = 0$$

Let us denote $\frac{a}{b} = C$, then

$$C^2 - C - 1 = 0$$

hence

$$C_1 = \frac{1 + \sqrt{5}}{2} = A, C_2 = \frac{1 - \sqrt{5}}{2}$$

Since we only consider a positive number, the golden ratio converges to 1.61803398874989... (Sen, Agarwal, 2008). The golden ratio can be simulated in the Matlab environment (Figure 2) with the following code (Mathworks, 2018):

```
>> n = 2:10;
>> ratio = fibonacci(n)./fibonacci(n - 1);
>> plot(n,ratio, '--o');
>> hold on
>> line(xlim, [1.618 1.618]);
>> hold off
```

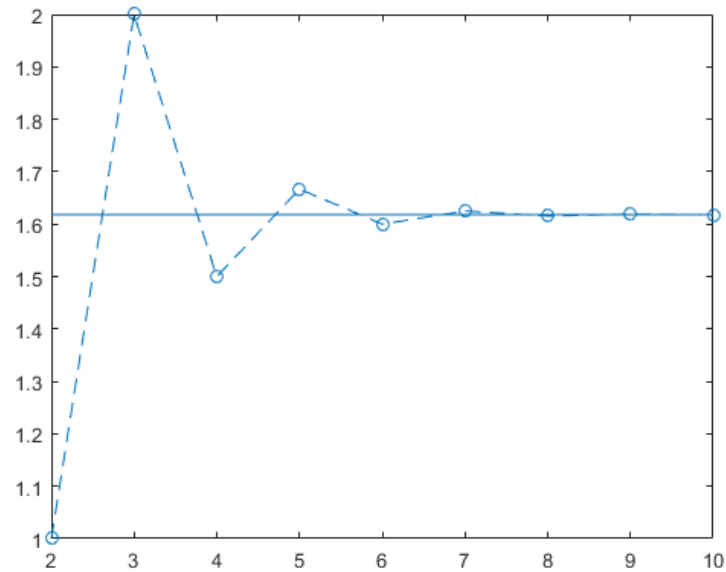


Fig. 2: Convergence to the golden ratio

Various specific geometric visualizations of the Fibonacci numbers and the golden ratio can be viewed, for example, in the paper (Weiss, Mick, 2014).

3 Fibonacci numbers and cryptography

Fibonacci numbers also have their justification in computer science and cryptography. (Sinha, 2017), (Elford, 2013) when encoding and decoding messages. Let us consider a message “word” and a password “abc”.

```
>> s = 'word';
>> pwd = 'abc';
```

We will encrypt this message using the Fibonacci sequence as follows:

1. Let us first rearrange the letters in the message inversely.


```
>> s = s(end:-1:1);
```

 By which we get the word „drow“
2. Now, we will combine the password „abc“ with the word we got to obtain the word „drowabc“.
3. Since our password has a length of 3, we will generate the first three numbers in the Fibonacci sequence: 1, 1, 2.


```
>> cnt = length(pwd);
>> f = fibonacci(cnt);
```
4. Now, the coding itself will consist in passing the entire length of the word „drowabc“ in the number of iterations how long the password (or the generated Fibonacci sequence) is and that each letter in the word is shifted ordinarily within the ASCII table:
 - a) for odd positions to the right by as many characters as determined by the element of the Fibonacci sequence in the current iteration
 - b) for even positions to the left by as many characters as determined by the element of the Fibonacci sequence in the current iteration

If we moved behind the letter „z“, while moving in the cycle to the right, we move up again starting from the letter „a“. On the contrary, if we moved before the letter „a“, while moving in the cycle to the left, we move down again to the letter „z“ (the alphabet of characters “a”...“z” is considered as if it were on a circle).

In our specific example, the word „drowabc“ would pass through three times. As a result, we got this coding sequence:

“eqpvbad”, “fpqucze”, “hnssexg”

with the word “**hnssexg**” being the encoding result; we have thus succeeded in encrypting the message “word” using the Fibonacci sequence for the password “abc”.

We implemented this algorithm into the Matlab computing environment as a batch file (script) and the entire source text looked as follows:

```
s = 'word';
pwd = 'abc';
s = s(end:-1:1);
s = [s pwd];
cnt = length(pwd);
f = fibonacci(cnt);
letters = ['a' 'b' 'c' 'd' 'e' 'f' 'g' 'h' 'i' 'j' 'k' 'l' 'm'
'n' 'o' 'p' 'q' 'r' 's' 't' 'u' 'v' 'w' 'x' 'y' 'z'];

for it = 1:cnt
    for jt = 1:length(s)
        index = find(letters == s(jt));
        if rem(jt, 2) == 1
            if index + f(it) > length(letters)
                fv = index + f(it) - length(letters);
            else
                fv = index + f(it);
            end
            s(jt) = letters(fv);
        else
            if index - f(it) < 1
                fv = length(letters) + (index - f(it));
            else
                fv = index - f(it);
            end
            s(jt) = letters(fv);
        end
    end
end
disp(s);
```

Decoding works on a precisely inverse principle, so we proceed in the cycle according to the generated Fibonacci sequence by the number of elements of the password length. For odd positions, we shift characters to the left, and for the even ones we do so to the right. If the number of password characters is different, the algorithm will not reach the original inverse

word combined with the password. Now, let us apply this algorithm to the given coded string “hnssexg” when we know the password “xyz”.

```
s = 'hnssexg';
pwd = 'abc';
cnt = length(pwd);
f = fibonacci(cnt);

letters = ['a' 'b' 'c' 'd' 'e' 'f' 'g' 'h' 'i' 'j' 'k' 'l' 'm'
'n' 'o' 'p' 'q' 'r' 's' 't' 'u' 'v' 'w' 'x' 'y' 'z'];
for it = 1:cnt
    for jt = 1:length(s)
        index = find(letters == s(jt));
        if rem(jt, 2) == 0
            if index + f(it) > length(letters)
                fv = index + f(it) - length(letters);
            else
                fv = index + f(it);
            end
            s(jt) = letters(fv);
        else
            if index - f(it) < 1
                fv = length(letters) + (index - f(it));
            else
                fv = index - f(it);
            end
            s(jt) = letters(fv);
        end
    end
end
disp(s);
```

If we had the correct password, the resulting string after having been decoded has been combined to the right, and we can select the decoded word to be inversely displayed:

```
if any(strfind(s, pwd)) == 1
    s = strrep(s, pwd, '');
    s = s(end:-1:1);
end
disp(s);
```

From the encoded string we got the original “word”.

4 Fibonacci numbers and school mathematics

Fibonacci numbers can also be included in teaching at primary school or within optional subjects, and they can be applied, for example, to teach the conversion of miles to kilometers and vice versa. This is an appropriate supplement and a matter of interest in teaching mathematics or physics using the Fibonacci sequence.

Let us consider a sequence of Fibonacci numbers (starting with the second element) 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, ... From physics, we know that 1 mile = 1,609344 kilometers. Let us now take any two consecutive numbers, e.g. 55 and 89, and we can assert that the first number in the sequence represents miles and the second one stands for kilometers (or vice versa).

```
>> mile = 55;
>> km = mile*1.609344
km = 88.5139
>> km = round(km)
km = 89
```

If we need to convert kilometers (or miles) that are not directly elements of the Fibonacci sequence, we first make a breakdown of that number into elements already present in the sequence (such a breakdown can always be made). Let us consider, for example, 15 miles that we need to convert to kilometers. The number 15 itself is not a element of the Fibonacci sequence, so we can first find the largest number within the sequence that does not exceed 15. It is 13, and we can make the decomposition in the way that $15 = 13 + 2$, with both elements being elements of the sequence. Based on the previous relationship, 13 miles = 21 km and 2 miles = 3 km. Then 15 miles = $21 + 3 = 24$ km. In fact,

```
>> km = round(15*1.609344)
km = 24
```

In the case of larger numbers containing series of ten, it is possible to proceed by taking out the series of ten first, converting a smaller number without a series of ten and then returning by multiplying by the original number. For example, if we needed to convert 1500 miles, it may be difficult to look for the appropriate elements of the Fibonacci sequence (unless we use the Matlab computing environment and the `fibonacci` function, for example).

However, as the number 1500 miles contains series of ten, namely 10^2 , we can first break down $1500 = 15 \cdot 10^2$. From the above, we know that 15 miles = 24 km.

Then $1500 = 15 \cdot 10^2$ miles = $24 \cdot 10^2$ km = 2400 km.

It should be noted, however, that the approximate value of the golden ratio is 1,618034, so for any two elements of the Fibonacci sequence, $a_n = 1,618034 \cdot a_{n-1}$ applies. While converting miles to kilometres, there is a constant factor defined as 1,609344. So, if we used the Fibonacci sequence to convert large numbers that have to be divided into multiple elements of the sequence where precision is lost when providing the conversion, then the distances would just be approximate.

Indeed, if we had a direct conversion of 1500 miles, we would get the exact distance of 2414 km and not 2400 km, which would be advisable to point out in teaching this principle. However, incorporating the Fibonacci principle into elementary school teaching or optional schooling may be of great interest and a way to get to know the possibilities of this issue.

Conclusion

Fibonacci numbers are an interesting issue, but despite their long history, they have not yet been scrutinized. We find Fibonacci numbers where we would not normally expect them to appear. Fibonacci numbers are found in evolution in different biological systems or in different

phenomena in nature, and for many mathematicians they represent beauty and perfection. They can be greatly used even today, such as being an indicator for trading with cryptocurrencies (Fischer, 2009).

Our aim in this paper was mainly to point out selected practical applications of Fibonacci numbers, such as their inclusion and use in cryptography and, on the other hand, their inclusion in the school environment where they can be used to teach conversions already at the lowest level of schools. The issue of Fibonacci numbers is appropriate to model and present, for example, by way of the Matlab computing environment, which proves to be a suitable tool for understanding different mathematical relationships and as a suitable environment for implementing various mathematical algorithms.

References

- Knott, R. - Quinney, D. A.: The life and numbers of Fibonacci. 1997, Available at: <http://pass.maths.org.uk/issue3/fibonacci/index.html>, N.d. Web. 2 February 2019.
- Fulier, J. – Šedivý, O.: Motivácia a tvorivosť vo vyučovaní matematiky. Nitra, UKF 2001, ISBN: 80-8050-445-8
- Weiss, G. – Mick, S.: Non-standard Visualizations of Fibonacci Numbers and the Golden Mean. KoG, Vol. 18. No. 18., Croatia 2014
- Znám, Š.: Teória čísel, Bratislava, SPN, 1975
- Dikovic, L.: Exploring Fibonacci Numbers using MATLAB. 10th International Scientific Conference, "Science and Higher Education in Function of Sustainable Development", 06 – 07 October 2017, Mečavnik – Drvengrad, Užice, Serbia
- Sen, S.K. - Agarwal, R. P.: Golden ratio in science, as random sequence source, its computation and beyond. Computers and Mathematics with Applications, Elsevier 2008
- Mathworks: online documentation, Available at: <https://www.mathworks.com/help/symbolic/fibonacci.html>, N.d. Web. 12 December 2018.
- Sinha, S.: The Fibonacci Numbers and Its Amazing Applications. International Journal of Engineering Science Invention, Volume 6 Issue 9, 2017, ISSN (Online): 2319 – 6734, ISSN (Print): 2319 – 6726
- Elfard, S. S.: Cryptography Based on the Linear Fibonacci Forms. In: Bulletin of University of Zawia, Vol . 2, No. 5, 2013
- Fischer, R.: Fibonacci Applications and Strategies for Traders. Wiley 2009, ISBN: 978-0471585206