BULLETIN OF THE ICA
Volume **103** (2025), 25–46

# A combinatorial proof for the Fibonacci dying rabbits problem

Roberto De Prisco

**Abstract.** We consider the generalized Fibonacci counting problem with rabbits that become fertile at age $f$ and die at age $d$, with $1 \leq f \leq d$, and $d$ finite or infinite. We provide a combinatorial proof of a recurrence relation for the number of rabbits at each generation. The proof is based exclusively on a counting argument and uses only elementary mathematics. The recurrence relation generalizes both the original Fibonacci sequence and several other Fibonacci-related sequences, such as the Padovan sequence and the Tribonacci, Tetranacci, and alike sequences.

## 1 Introduction

Leonardo Bonacci, better known as Fibonacci, among his many contributions to the field, considered the well known rabbits counting problem that resulted in the so-called Fibonacci sequence of integers. The problem is the following: A population of pairs of rabbits, starting with one newborn pair, grows in each generation with every fertile pair of rabbits giving birth to a new pair of rabbits. A pair of rabbits becomes fertile at age 2; that is, it does not proliferate in the generation in which it is born, but starts proliferating in the next generation, and the newborns are added the subsequent generation. The problem is that of counting the number $F_n$ of pairs of rabbits at every generation $n \geq 1$. The initial condition gives $F_1 = 1$. The unique pair of rabbits is not fertile in the first generation, and thus $F_2 = 1$. In the second generation, the pair proliferates giving birth to a newborn pair of rabbits, which is added to the subsequent generation, and thus $F_3 = 2$. For the next generation only the initial pair of rabbits proliferates and thus $F_4 = 3$. For the subsequent one, there are two pairs of rabbits that proliferate, and thus $F_5 = 5$. Proceeding in this way one gets

**Key words and phrases:** Fibonacci, Generalized Fibonacci Sequences, Dying Rabbits Problem
**AMS (MOS) Subject Classifications:** 11B39, 11K31

Received: 5 January 2024
Accepted: 6 April 2024

the so-called Fibonacci sequence

$$1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, \ldots.$$

The well known recurrence relation

$$F_n = F_{n-1} + F_{n-2} \tag{1}$$

gives an easy way to compute the Fibonacci sequence, for any $n \geq 3$, starting from the initial conditions $F_1 = 1$ and $F_2 = 1$. One can extend the sequence by adding $F_0 = 0$, and the formula is still valid for $n \geq 2$, with the initial conditions $F_0 = 0$ and $F_1 = 1$. Binet's closed formula provides a direct way of computing $F_n = \frac{1}{\sqrt{5}}\left[\left(\frac{1+\sqrt{5}}{2}\right)^n - \left(\frac{1-\sqrt{5}}{2}\right)^n\right]$. It is well know that the Fibonacci sequence has many interesting properties that have led to its fame.

A number of generalizations have been considered. For example the Lucas sequence, is obtained by changing the initial conditions to $F_0 = 2$ and $F_2 = 1$. The Gibonacci sequence is a further generalization for which $F_0$ and $F_1$ can be arbitrary; thus Fibonacci and Lucas numbers are special cases of Gibonacci numbers.

The Padovan sequence is defined as $F_n = F_{n-2} + F_{n-3}$, with initial values $F_0 = F_1 = F_2 = 1$, and has properties similar to those of the Fibonacci sequence.

Another generalization, that has been widely studied, is the $k$-step Fibonacci sequence [8], where an element of the sequence is obtained by adding the previous $k$ elements, that is, $F_n = \sum_{i=1}^{k} F_{n-i}$. For the case of 3 or 4 terms, the sequences are called Tribonacci and Tetranacci numbers (see also [9] for further details). A closed formula for this generalization is given in [3].

There are other generalizations. Plenty of research papers and several books have been written about the Fibonacci numbers, their many properties and their generalizations; we refer the reader to [7, 10] for more information.

The generalizations that we cited above consider a direct modification of the recurrence relation $F_n = F_{n-1} + F_{n-2}$. In this paper we consider the generalization in which, in the original problem, cast as a counting problem of a growing population of rabbits, the rabbits become fertile after some number $f$ of generations and at some point, after $d$ generations, they die.

This problem is also called the dying rabbits problem and has been studied in several papers [1,2,4–6,11]. This specific generalization is equivalent to some generalizations that directly change the recurrence relation.

**Contribution of this paper.** We provide a combinatorial proof for a recurrence relation that gives the $n^{\text{th}}$ generalized Fibonacci number as a function of 2 or 3 previous numbers. More specifically, we prove that, with the initial condition $F_1 = 1$, the number of rabbits for the $n^{\text{th}}$ generation is given by

$$F_n = \begin{cases} 1, & \text{for} \quad 2 \leq n \leq f \quad \text{(case 1)}, \\ F_{n-1} + F_{n-f}, & \text{for} \ f+1 \leq n \leq d \ \text{(case 2)}, \\ F_{n-1} + F_{n-f} - 1, & \text{for} \quad n = d+1 \quad \text{(case 3)}, \\ F_{n-1} + F_{n-f} - F_{n-d-1}, & \text{for} \quad n \geq d+2 \quad \text{(case 4)}. \end{cases}$$

This formula clearly generalizes Equation (1). Indeed the Fibonacci sequence is the case for which $f = 2$ and $d = \infty$, and for this choice of the two parameters, we only have cases 1 and 2. For $n = 2$, we have case 1, which gives $F_2 = 1$, and for $n \geq 3$, case 2 becomes $F_n = F_{n-1} + F_{n-2}$. The formula generalizes also several other well known Fibonacci-like sequences, as we will point out in Section 4.6. We remark that the problem of counting the rabbits for the dying rabbits problem has been solved in [6] (more details in Section 2). However, the recurrence relation that we propose in this paper has not been explicitly given in previous papers. Moreover, the proof that we provide is quite simple and uses only elementary mathematics.

**Paper organization.** In Section 2 we cite and compare relevant previous work. Then, in Section 3 we provide some basic observations and the "base equation", and in Section 4 we give the proof of the recurrence relation by "unraveling" the base equation. Section 5 closes the paper with a brief conclusion. In Appendix B we give a numerical example, and Appendix A contains a Java program that counts the rabbits through a simulation of the evolution of the population.

## 2 Previous work

The earliest papers, that we are aware of, about the Fibonacci counting problem with dying rabbits, are by Brother U. Alfred who, in [1], posed the question of counting the rabbits for the specific case $f = 2$ and $d = 12$, seemingly thinking that it was a relatively easy counting problem, and later, in [2], concluded that the problem did not seem that easy. A recurrence relation that correctly solves the case $f = 2$, $d = 12$, was provided by Cohn [4]. The recurrence relation matches the general formula that we provide in this paper.

Hoggatt and Lind [5, 6] considered a generalization on the breeding pattern: Each pair of rabbits breeds $B_1$ new pairs in its first generation, $B_2$ in the second and so on, with $B_0 = 0$; rabbits die after a fixed number $d$ of generations. In [6] the authors provided a solution in terms of the polynomials associated to the birth and death patterns. More specifically, denoting with $B(x) = \sum_{n\geq 0} B_n x^n$ the polynomial associated to the birth pattern and with $D(x) = x^d$ the one associated to the deaths, the solution provided in [6] is

$$F(x) = \frac{1 - D(x)}{(1 - x)(1 - B(x))}.$$

From this equation it is possible to get a recurrence relation for specific birth and death patterns. However no general recurrence relation is provided in [6], neither for the case we are considering, that is for the pattern $B_n = 0$ for $0 \leq n \leq f - 1$ and $B_n = 1$ for $n \geq f$ and $D(x) = x^d$, nor for other cases—with the exception of two examples, one of which is the specific case considered in [1].

For the specific problem that we consider in this paper, Oller-Marcén [11] provides a recurrence relation, namely[1]

$$F_n = \begin{cases} 1, & \text{for } 1 \leq n \leq f, \\ F_{n-1} + F_{n-f}, & \text{for } f < n \leq d, \\ F_{n-f} + F_{n-f-1} + \cdots + F_{n-d}, & \text{for } n > d. \end{cases} \tag{2}$$

The proof of the above equation provided in [11], in Proposition 9, is quite short, and although the first two cases are easy and clear, the third case, the one for $n > d$, is not as much clear[2].

[1]The notation used in [11] uses $h$ for the age in which rabbits become fertile and $k$ for the generations that rabbits live *after* the fertile age. Appropriately matching the different notations, the correspondence is $f = h$ and $d = k + h - 1$.

[2]Literally—adjusting only the terminology and the notation to make them conform to the ones we are using in this paper—the proof states, “The number of rabbits at the $n^{\text{th}}$ generation can be computed as the sum of all the preceding rabbits except those which are not mature yet ($F_{n-j}$, with $1 \leq j \leq f-1$) and those which have died ($F_{n-j}$, with $j > n - d$)”. Although this sentence describes the right side of the third case of Equation (2), it is not clear why it should give the total number of rabbits for the $n^{\text{th}}$ generation. It is not true that the number of rabbits that are not mature in generation $n$ is equal to $F_{n-1} + F_{n-2} + \cdots + F_{n-f+1}$, as this sum is the sum of all the alive rabbits in a number of generations, and thus non-mature rabbits would be counted multiple

The approach used in the proof that we provide in this paper, leading to a more compact formula, can probably be used to explain also (2), which in fact is equivalent to the one we propose, as we show in Section 4.6. In [11] the solution is studied also as a function of the roots of the characteristic polynomial.

# 3 The "base equation"

First, to ease the wording, we will talk about single rabbits instead of pairs of rabbits. A single rabbit that proliferates is not natural[3], but considering single rabbits instead of pairs of rabbits does not change the underlying counting problem. Let $F_n$ be the number of rabbits at the $n^{\text{th}}$ generation, with the initial condition $F_1 = 1$. The rabbits become fertile at the $f^{\text{th}}$ generation and die at the age of $d$, with $1 \leq f \leq d$ (the case $d < f$ is trivial since the single initial rabbit will die before proliferating).

Rabbits with age $d$ first proliferate and then die. A newborn rabbit has age 1. The "step of the evolution" is as follows: Given a population $F_n$, every element of the population with age at least $f$ gives birth to a new element for the next population; then the age of each element is increased by 1, and rabbits with age $> d$ die and thus will not be part of the next generation.

The Fibonacci sequence is the special case $f = 2$ and $d = \infty$, in which the rabbits become fertile after their first generation, that is, at the second generation, and never die.

Consider the $n^{\text{th}}$ generation and define $newborns_n$ and $deaths_n$ as the number of, respectively, newborn rabbits and deaths that affect the population of this generation. Notice that $newborns_n$ is equal to the number of fertile rabbits in the previous $(n-1)^{\text{th}}$ generation and $deaths_n$ is equal to the number of rabbits that have exactly age $d$ in the previous $(n-1)^{\text{th}}$ generation—we remark that these rabbits were alive in generation $n-1$, as they die at the end of generation $n-1$. The following basic fact is immediate.

**Base equation.** The number of rabbits at the $n^{\text{th}}$ generation is equal to the number of rabbits in the previous generation plus the number of newborns

times. Moreover, some fertile rabbits, alive in the previous generations, might have died before generation $n$. A similar remark, about the multiple counting, applies to the number of rabbits that have died.

[3] Also the setting of the original problem is not very natural!

minus the number of deaths, that is,

$$F_n = F_{n-1} + newborns_n - deaths_n. \tag{3}$$

We call the above equation the *base equation.* Once we are able to evaluate the newborns and the deaths for each generation, the base equation gives, in a straightforward way, the solution.

Before we proceed, let us make some observations. The degenerate case $d < f$ would give rise to the sequence

$$\underbrace{1, 1, \ldots, 1}_{d \text{ times}}, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, \ldots$$

since the unique rabbit dies before proliferating. Thus, it is not interesting, and this is why we consider only the case $f \leq d$.

The borderline case $f = d$, is simple to deal with. Indeed, in this case every rabbit proliferates in the same generation in which it dies, that is, $newborns_n = deaths_n$ (either 0 for 1). Hence, the total number of rabbits never changes. Thus, the sequence that we get is

$$1, 1, 1, 1, 1, 1, 1, 1, 1, 1, \ldots.$$

When $f < d$, evaluating the number of newborns and deaths becomes trickier; however, if $d = \infty$ evaluating the deaths is easy: $deaths_n = 0$.

The special case $f = 1$ (and $d = \infty$), for which rabbits are immediately fertile, leads to a doubling of the rabbits at every generation. Indeed we would have $newborns_n = F_{n-1}$, leading to $F_n = 2F_{n-1}$, which gives the sequence of powers of 2:

$$1, 2, 4, 8, 16, 32, 64, 128, 256, 1024, \ldots.$$

For the original Fibonacci sequence, beside $d = \infty$, we have $f = 2$, and also in this case it is easy to evaluate the newborns: Among the previous population of size $F_{n-1}$ there are exactly $F_{n-2}$ fertile elements, since rabbits become fertile at age 2, and thus there are exactly $F_{n-2}$ newborns for the new generation, that is, $newborns_n = F_{n-2}$. Thus for the case $f = 2$, $d = \infty$, we have the well known Fibonacci's formula $F_n = F_{n-1} + F_{n-2}$.

More in general, for the case $d = \infty$ and any finite $f$, we have that the number of rabbits that are fertile for generation $n$ is exactly $F_{n-f}$, that is, all the rabbits that were in the population $f$ generations before; indeed all these rabbits have age at least $f$ in generation $n$ and they have not died.

All the other rabbits are still too young. The number of fertile rabbits gives the number of newborns, that is, $newborns_n = F_{n-f}$. This results in the recurrence relation $F_n = F_{n-1}+F_{n-f}$, as also stated in Equation (2).

However, when $d$ is finite, evaluating the exact number of newborns and deaths seems trickier. Indeed it is not true anymore that the number of fertile rabbits for the $n^{\text{th}}$ generation is equal to $F_{n-f}$ because some of those rabbits could have died meanwhile. Also the total number of deaths seems more difficult to assess.

# 4 Unraveling the base equation

Keeping track only of the total number of rabbits in each generation does not help that much. Counting also the number of rabbits for each possible age allows to clearly define the relations among the numbers that we get from the counting. In the following we first provide some basic definitions and properties. Then, exploiting such properties, we prove the four cases of the proposed recurrence relation.

## 4.1 Definitions and properties

We start with the following definition for the number of rabbits with a specific age in a given generation.

**Definition 4.1.** Define $F_n^x$, for $x = 1, 2, \ldots, d$ as the number of rabbits of age $x$ at (the beginning of) generation $n$.

In the following, we study the relation among all the $F_n^x$, for any $n$ and $x$, and the total number of rabbits in each generation, that is, $F_n$, for any $n \geq 1$.

**Example.** In order to clarify the theoretical analysis we will instantiate each relation with an example. In Appendix B, Table B.1, we provide a table that reports the values for the specific case of $f = 3$ and $d = 9$ up to $n = 35$, and in the following we will refer to pieces of that table to provide the examples.

We start with an obvious relation which follows directly from the definition of the $F_n^x$.

$$F_n = F_n^1 + F_n^2 + \cdots + F_n^{d-1} + F_n^d. \tag{4}$$

**Example.** Below is the row for the 14th generation from Table B.1. The total number of rabbits is $F_{14} = 79$ and among these 79 rabbits, 25 are newborns (that is, with age 1), 18 with age 2, 12 with age 3 and so on, up to just 1 rabbit with age 9.

| $n$ | $F_n$ | $F_n^1$ | $F_n^2$ | $F_n^3$ | $F_n^4$ | $F_n^5$ | $F_n^6$ | $F_n^7$ | $F_n^8$ | $F_n^9$ |
|---|---|---|---|---|---|---|---|---|---|---|
| 14 | **79** | 25 | 18 | 12 | 8 | 6 | 4 | 3 | 2 | 1 |

We have that $79 = 25 + 18 + 12 + 8 + 6 + 4 + 3 + 2 + 1$.

**Lemma 4.2.** *For any $x \leq \min\{d, n\}$, we have that $F_n^x = F_{n-1}^{x-1} = F_{n-2}^{x-2} = \cdots = F_{n-x+1}^1$.*

*Proof.* The age of the rabbits increases by 1 at each generation. Thus, if there are $F_n^x$ rabbits (of age $x$) at generation $n$, there must have been the same number $F_{n-1}^{x-1}$ (of age $x-1$) at generation $n-1$, and the same number $F_{n-2}^{x-2}$ (of age $x-2$) at generation $n-2$ and so on, up to generation $n-x+1$, in which the rabbits were newborns. The condition $x \leq \min\{d, n\}$ ensures that $F_n^x$ is defined and that $n-x+1 \geq 1$ refers to an existing generation. □

**Example.** Below are the rows from the 12th through the 20th generation from Table B.1. The $F_{12}^1 = 12$ rabbits with age 1 in generation 12 will have age 2 in generation 13 (where they are counted as $F_{13}^2$), age 3 in generation 14 ($F_{14}^3$), and so on, up to age 9 in generation 20 ($F_{12}^9$). Then, they die and will not be part of generation 21.

| $n$ | $F_n$ | $F_n^1$ | $F_n^2$ | $F_n^3$ | $F_n^4$ | $F_n^5$ | $F_n^6$ | $F_n^7$ | $F_n^8$ | $F_n^9$ |
|---|---|---|---|---|---|---|---|---|---|---|
| 12 | 38 | **12** | 8 | 6 | 4 | 3 | 2 | 1 | 1 | 1 |
| 13 | 55 | 28 | **12** | 8 | 6 | 4 | 3 | 2 | 1 | 1 |
| 14 | 79 | 25 | 28 | **12** | 8 | 6 | 4 | 3 | 2 | 1 |
| 15 | 114 | 36 | 25 | 28 | **12** | 8 | 6 | 4 | 3 | 2 |
| 16 | 165 | 53 | 36 | 25 | 28 | **12** | 8 | 6 | 4 | 3 |
| 17 | 238 | 76 | 53 | 36 | 25 | 28 | **12** | 8 | 6 | 4 |
| 18 | 343 | 109 | 76 | 53 | 36 | 25 | 28 | **12** | 8 | 6 |
| 19 | 495 | 158 | 109 | 76 | 53 | 36 | 25 | 28 | **12** | 8 |
| 20 | 715 | 228 | 158 | 109 | 76 | 53 | 36 | 25 | 28 | **12** |

Graphically, Lemma 4.2, implies that all numbers $F_n^x$ move diagonally in the table, down-right when going to the next generation, up-left when going back to the previous generation.

**Lemma 4.3.** *For $n \geq 2$, the number of rabbits that die at generation $n$, that is, the ones that are alive in generation $n-1$ but dead for generation $n$, is $F^d_{n-1}$.*

*Proof.* Immediate from Definition 4.1: $F^d_{n-1}$ is the number of rabbits with age $d$. The condition $n \geq 2$ ensures that $F^d_{n-1}$ is defined. □

**Example.** The row for the 34th generation of Table B.1 is shown below.

| $n$ | $F_n$ | $F^1_n$ | $F^2_n$ | $F^3_n$ | $F^4_n$ | $F^5_n$ | $F^6_n$ | $F^7_n$ | $F^8_n$ | $F^9_n$ |
|---|---|---|---|---|---|---|---|---|---|---|
| 34 | 121644 | 38788 | 26875 | 18621 | 12902 | 8939 | 6194 | 4292 | 2973 | **2060** |

The total number of rabbits with age 9 is $F^9_{34} = 2060$. These are the rabbits that die at the end of generation 34 and thus will not be part of the 35th generation.

**Lemma 4.4.** *The number $F^1_n$ of newborn rabbits at generation $n$ is equal to $\sum_{x=f}^{d} F^x_{n-1}$.*

*Proof.* The number of newborns is equal to the number of fertile rabbits in the previous generation. Thus we need to consider all the rabbits that in the previous generation have age at least $f$. By Definition 4.1 we have that there are $F^f_{n-1}$ rabbits of age $f$, $F^{f+1}_{n-1}$ rabbits of age $f+1$, and so on up to $F^d_{n-1}$ rabbits of age $d$. Thus the total number of rabbits that have a fertile age, that is, an age at least $f$, in generation $n-1$, is $F^f_{n-1}+F^{f+1}_{n-1}+\cdots+F^d_{n-1}$. □

**Example.** Below are shown the rows for the 25th and the 26th generations from Table B.1.

| $n$ | $F_n$ | $F^1_n$ | $F^2_n$ | $F^3_n$ | $F^4_n$ | $F^5_n$ | $F^6_n$ | $F^7_n$ | $F^8_n$ | $F^9_n$ |
|---|---|---|---|---|---|---|---|---|---|---|
| 25 | 4477 | 1428 | 989 | **685** | **475** | **329** | **228** | **158** | **109** | **76** |
| 26 | 6461 | **2060** | 1428 | 989 | 685 | 475 | 329 | 228 | 158 | 109 |

The newborn rabbits for generation 26 are $F^1_{26} = 2060$, and since $f = 3$, we have $2060 = 685 + 475 + 329 + 228 + 158 + 109 + 76$.

Now, armed with the above equalities, we can easily unravel the base equation. We distinguish four subcases.

### 4.2 Case 1: $2 \leq n \leq f$

In the first $f$ generations there are no newborns nor deaths. Hence we have $newborns_n = deaths_n = 0$, and since $F_1 = 1$, the base Equation (3) reduces to

$$F_n = 1, \qquad \text{for } 2 \leq n \leq f. \tag{5}$$

### 4.3 Case 2: $f + 1 \leq n \leq d$

Since $n \leq d$, there are no deaths. Thus $deaths_n = 0$. However, some rabbits start to proliferate. By Lemma 4.4 we have that the newborns for generation $n$ are $F_n^1 = \sum_{x=f}^{d} F_{n-1}^x$.

By Lemma 4.2 we have that each element $F_{n-1}^x$ of the sum can be substituted by the equal term $F_{n-1-(f-1)}^{x-(f-1)}$, which we find by going up-left diagonally for $f-1$ generations, and thus, starting from the base equation, we have that

$$\begin{aligned}
F_n &= F_{n-1} + newborns_n - deaths_n \\
&= F_{n-1} + F_n^1 - 0 \\
&= F_{n-1} + \sum_{x=f}^{d} F_{n-1}^x && \text{(by Lemma 4.4)} \\
&= F_{n-1} + \sum_{x=f}^{d} F_{n-f}^{x-f+1} && \text{(by Lemma 4.2)} \\
&= F_{n-1} + \sum_{x=1}^{d-f+1} F_{n-f}^x && \text{(index substitution)} \\
&= F_{n-1} + \sum_{x=1}^{d} F_{n-f}^x && \text{(added elements are 0)} \\
&= F_{n-1} + F_{n-f} && \text{(by Equation (4))}
\end{aligned}$$

where the second-to-last step is true because for generation $n - f$ there are no rabbits with age $x > d - f$ since the condition $n \leq d$ implies that $n - f \leq d - f$ and thus no rabbit can have an age bigger than this number. Thus $F_{n-1}^x = 0$ for $x > d - f$.

**Example.** $F_9 = F_8+F_8^3+F_8^4+F_8^5+F_8^6+F_8^7+F_8^8+F_8^9$, that is $13 = 9+1+1+1+0+0+1+0$, which is also equal to $F_8+F_6^1+F_6^2+F_6^3+F_6^4+F_6^5+F_6^6+F_6^7$, and also to $F_8 + \sum x = 1^9 F_6^x$ since the added elements $F_6^8$ and $F_6^9$ are 0.

| $n$ | $F_n$ | $F_n^1$ | $F_n^2$ | $F_n^3$ | $F_n^4$ | $F_n^5$ | $F_n^6$ | $F_n^7$ | $F_n^8$ | $F_n^9$ |
|---|---|---|---|---|---|---|---|---|---|---|
| 6 | **4** | **1** | **1** | **1** | **0** | **0** | **1** | **0** | 0 | 0 |
| 7 | 6 | 2 | **1** | **1** | **1** | **0** | **0** | **1** | **0** | 0 |
| 8 | **9** | 3 | 2 | **1** | **1** | **1** | **0** | **0** | **1** | **0** |
| 9 | **13** | 4 | 3 | 2 | 1 | 1 | 1 | 0 | 0 | 1 |

Hence we have

$$F_n = F_{n-1} + F_{n-f}, \qquad \text{for } f < n \leq d. \tag{6}$$

## 4.4 Case 3: $n = d + 1$

This case is very similar to the previous one with the unique exception that we need to account for the first death: Indeed, in generation $d$ the first rabbit dies, and it is the only one that dies. Thus we have that $deaths_n = F_{n-1}^d = F_d^d = 1$. The analysis of the newborns carried out for the previous case applies also to this case. Hence we have that

$$F_{d+1} = F_d - F_{d+1-f} - 1. \tag{7}$$

## 4.5 Case 4: $n \geq d + 2$

Next, we unravel Equation (3) for the other values of $n$. The reasoning that we will provide for this case cannot be applied to the previous ones because it involves $F_{n-d-1}$, which is not defined for $n < d+2$. We have that

$$\begin{aligned} F_n &= F_{n-1} + newborns_n - deaths_n \\ &= F_{n-1} + F_n^1 - F_{n-1}^d \\ &= F_{n-1} + \sum_{x=f}^{d} F_{n-1}^x - F_{n-1}^d \qquad \text{(by Lemma 4.4)} \\ &= F_{n-1} + \sum_{x=f}^{d-1} F_{n-1}^x \end{aligned}$$

**Example.** $F_{35} = F_{34} + F_{34}^3 + F_{34}^4 + F_{34}^5 + F_{34}^6 + F_{34}^7 + F_{34}^8$, that is 175565 = 121644 + 18621 + 12902 + 8939 + 6194 + 4292 + 2973.

| $n$ | $F_n$ | $F_n^1$ | $F_n^2$ | $F_n^3$ | $F_n^4$ | $F_n^5$ | $F_n^6$ | $F_n^7$ | $F_n^8$ | $F_n^9$ |
|---|---|---|---|---|---|---|---|---|---|---|
| 34 | **121644** | 38788 | 26875 | **18621** | **12902** | **8939** | **6194** | **4292** | **2973** | 2060 |
| 35 | **175565** | 55981 | 38788 | 26875 | 18621 | 12902 | 8939 | 6194 | 4292 | 2973 |

As done for case 2, by Lemma 4.2 we have that each element $F_{n-1}^x$ of the sum can be substituted by the equal term $F_{n-1-(f-1)}^{x-(f-1)}$, which we find by going up-left diagonally for $f-1$ generations, and thus we have that

$$\begin{aligned} F_n &= F_{n-1} + \sum_{x=f}^{d-1} F_{n-1}^x \\ &= F_{n-1} + \sum_{x=f}^{d-1} F_{n-f}^{x-f+1} && \text{(by Lemma 4.2)} \\ &= F_{n-1} + \sum_{x=1}^{d-f} F_{n-f}^x && \text{(index substitution)} \end{aligned}$$

**Example.** $F_{35} = F_{34} + F_{32}^1 + F_{32}^2 + F_{32}^3 + F_{32}^4 + F_{32}^5 + F_{32}^6$, that is 175565 = 121644 + 18621 + 12902 + 8939 + 6194 + 4292 + 2973.

| $n$ | $F_n$ | $F_n^1$ | $F_n^2$ | $F_n^3$ | $F_n^4$ | $F_n^5$ | $F_n^6$ | $F_n^7$ | $F_n^8$ | $F_n^9$ |
|---|---|---|---|---|---|---|---|---|---|---|
| 32 | 58398 | **18621** | **12902** | **8939** | **6194** | **4292** | **2973** | 2060 | 1428 | 989 |
| 33 | 84284 | 26875 | **18621** | **12902** | **8939** | **6194** | **4292** | **2973** | 2060 | 1428 |
| 34 | **121644** | 38788 | 26875 | **18621** | **12902** | **8939** | **6194** | **4292** | **2973** | 2060 |
| 35 | **75565** | 55981 | 38788 | 26875 | 18621 | 12902 | 8939 | 6194 | 4292 | 2973 |

We now observe that

$$F_{n-f} = \sum_{x=1}^{d} F_{n-f}^x = \sum_{x=1}^{d-f} F_{n-f}^x + \sum_{x=d-f+1}^{d} F_{n-f}^x,$$

and thus we have that

$$\sum_{x=1}^{d-f} F_{n-f}^x = F_{n-f} - \sum_{x=d-f+1}^{d} F_{n-f}^x.$$

Hence, we have that

$$F_n = F_{n-1} + \sum_{x=1}^{d-f} F_{n-f}^x = F_{n-1} + F_{n-f} - \sum_{x=d-f+1}^{d} F_{n-f}^x.$$

**Example.** $F_{35} = F_{34} + F_{32} - F_{32}^7 - F_{32}^8 - F_{32}^9$, that is $175565 = 121644 + 58398 - 2060 - 1428 - 989$.

| $n$ | $F_n$ | $F_n^1$ | $F_n^2$ | $F_n^3$ | $F_n^4$ | $F_n^5$ | $F_n^6$ | $F_n^7$ | $F_n^8$ | $F_n^9$ |
|---|---|---|---|---|---|---|---|---|---|---|
| 32 | **58398** | 18621 | 12902 | 8939 | 6194 | 4292 | 2973 | **2060** | **1428** | **989** |
| 33 | 84284 | 26875 | 18621 | 12902 | 8939 | 6194 | 4292 | 2973 | 2060 | 1428 |
| 34 | **121644** | 38788 | 26875 | 18621 | 12902 | 8939 | 6194 | 4292 | 2973 | 2060 |
| 35 | **175565** | 55981 | 38788 | 26875 | 18621 | 12902 | 8939 | 6194 | 4292 | 2973 |

As done before we can now shift up-left for $d - f$ generations the terms of the sum, using Lemma 4.2:

$$\sum_{x=d-f+1}^{d} F_{n-f}^x = \sum_{x=d-f+1-(d-f)}^{d-(d-f)} F_{n-d}^x \qquad \text{(by Lemma 4.2)}$$

$$= \sum_{x=1}^{f} F_{n-d}^x \qquad \text{(index substitution)}$$

Thus we have that

$$F_n = F_{n-1} + F_{n-f} - \sum_{x=d-f+1}^{d} F_{n-f}^x = F_{n-1} + F_{n-f} - \sum_{x=1}^{f} F_{n-d}^x. \tag{8}$$

**Example.** First, we have that $\sum_{x=7}^{9} F_{32}^x = \sum_{x=1}^{3} F_{26}^x$, since the three terms are equal (they have just moved up-left diagonally for 6 generations. Then we have $F_{35} = F_{34} + F_{32} - F_{26}^1 - F_{26}^2 - F_{26}^3$, that is $175565 = 121644 + 58398 - 2060 - 1428 - 989$.

| $n$ | $F_n$ | $F_n^1$ | $F_n^2$ | $F_n^3$ | $F_n^4$ | $F_n^5$ | $F_n^6$ | $F_n^7$ | $F_n^8$ | $F_n^9$ |
|---|---|---|---|---|---|---|---|---|---|---|
| 26 | 6461 | **2060** | **1428** | **989** | 685 | 475 | 329 | 228 | 158 | 109 |
| 27 | 9325 | 2973 | **2060** | **1428** | **989** | 685 | 475 | 329 | 228 | 158 |
| 28 | 13459 | 4292 | 2973 | **2060** | **1428** | **989** | 685 | 475 | 329 | 228 |
| 29 | 19425 | 6194 | 4292 | 2973 | **2060** | **1428** | **989** | 685 | 475 | 329 |
| 30 | 28035 | 8939 | 6194 | 4292 | 2973 | **2060** | **1428** | **989** | 685 | 475 |
| 31 | 40462 | 12902 | 8939 | 6194 | 4292 | 2973 | **2060** | **1428** | **989** | 685 |
| 32 | **58398** | 18621 | 12902 | 8939 | 6194 | 4292 | 2973 | **2060** | **1428** | **989** |
| 33 | 84284 | 26875 | 18621 | 12902 | 8939 | 6194 | 4292 | 2973 | 2060 | 1428 |
| 34 | **121644** | 38788 | 26875 | 18621 | 12902 | 8939 | 6194 | 4292 | 2973 | 2060 |
| 35 | **175565** | 55981 | 38788 | 26875 | 18621 | 12902 | 8939 | 6194 | 4292 | 2973 |

And to conclude the unraveling of the formula, we observe that the sum of the negative terms $F^x_{n-d}$ is equal to the total number of rabbits $F_{n-d-1}$ in the previous generation. Indeed, the first term $F^1_{n-d}$ of this sum is the number of newborns of generation $n-d$, which is equal to the number of fertile rabbits in generation $n-d-1$; that is,

$$F^1_{n-d} = \sum_{x=f}^{d} F^x_{n-d-1}$$

and the other terms, shifting them up-left of 1 generation, using again Lemma 4.2, are equal to

$$\sum_{x=2}^{f} F^x_{n-d} = \sum_{x=1}^{f-1} F^x_{n-d-1}.$$

By putting together these facts we have that

$$\sum_{x=1}^{f} F^x_{n-d} = F^1_{n-d} + \sum_{x=2}^{f} F^x_{n-d} = \sum_{x=f}^{d} F^x_{n-d-1} + \sum_{x=1}^{f-1} F^x_{n-d-1} = F_{n-d-1}. \quad (9)$$

**Example.** $F^1_{26} + F^2_{26} + F^3_{26} = F_{25}$, that is $2060 + 1428 + 989 = 4477$, because $F^1_{26} = F^3_{25} + F^4_{25} + F^5_{25} + F^6_{25} + F^7_{25} + F^8_{25} + F^9_{25}$, that is $2060 = 685 + 475 + 329 + 26 + 228 + 158 + 109 + 76$ and the two missing terms are equal to the two terms that come from the next generation, that is $F^1_{25} = F^2_{26} = 1428$ and $F^2_{25} = F^3_{26} = 989$.

| $n$ | $F_n$ | $F_n^1$ | $F_n^2$ | $F_n^3$ | $F_n^4$ | $F_n^5$ | $F_n^6$ | $F_n^7$ | $F_n^8$ | $F_n^9$ |
|---|---|---|---|---|---|---|---|---|---|---|
| 25 | **4477** | **1428** | **989** | **685** | **475** | **329** | **228** | **158** | **109** | **76** |
| 26 | 6461 | **2060** | **1428** | **989** | 685 | 475 | 329 | 228 | 158 | 109 |

Putting together Equations (8) and (9) we have that

$$F_n = F_{n-1} + F_{n-f} - F_{n-d-1}, \qquad \text{for } n \geq d+2. \tag{10}$$

**Example.** $F_{35} = F_{34} + F_{32} - F_{26}$, that is $175565 = 121644 + 58398 - 4477$.

| $n$ | $F_n$ | $F_n^1$ | $F_n^2$ | $F_n^3$ | $F_n^4$ | $F_n^5$ | $F_n^6$ | $F_n^7$ | $F_n^8$ | $F_n^9$ |
|---|---|---|---|---|---|---|---|---|---|---|
| 25 | **4477** | 1428 | 989 | 685 | 475 | 329 | 228 | 158 | 109 | 76 |
| 26 | 6461 | 2060 | 1428 | 989 | 685 | 475 | 329 | 228 | 158 | 109 |
| 27 | 9325 | 2973 | 2060 | 1428 | 989 | 685 | 475 | 329 | 228 | 158 |
| 28 | 13459 | 4292 | 2973 | 2060 | 1428 | 989 | 685 | 475 | 329 | 228 |
| 29 | 19425 | 6194 | 4292 | 2973 | 2060 | 1428 | 989 | 685 | 475 | 329 |
| 30 | 28035 | 8939 | 6194 | 4292 | 2973 | 2060 | 1428 | 989 | 685 | 475 |
| 31 | 40462 | 12902 | 8939 | 6194 | 4292 | 2973 | 2060 | 1428 | 989 | 685 |
| 32 | **58398** | 18621 | 12902 | 8939 | 6194 | 4292 | 2973 | 2060 | 1428 | 989 |
| 33 | 84284 | 26875 | 18621 | 12902 | 8939 | 6194 | 4292 | 2973 | 2060 | 1428 |
| 34 | **121644** | 38788 | 26875 | 18621 | 12902 | 8939 | 6194 | 4292 | 2973 | 2060 |
| 35 | **175565** | 55981 | 38788 | 26875 | 18621 | 12902 | 8939 | 6194 | 4292 | 2973 |

## 4.6 The recurrence relation

To summarize, we provide the following theorem.

**Theorem 4.5.** *Let $F_1 = 1$ and let $f$ and $d$ be integers such that $1 \leq f \leq d$. The number $F_n$ of rabbits at generation $n$, for a population of rabbits that become fertile at age $f$ and die at age $d \geq f$, is given by*

$$F_n = \begin{cases} 1, & \text{for } 2 \leq n \leq f \\ F_{n-1} + F_{n-f}, & \text{for } f < n \leq d \\ F_{n-1} + F_{n-f} - 1, & \text{for } n = d+1 \\ F_{n-1} + F_{n-f} - F_{n-d-1}, & \text{for } n \geq d+2. \end{cases} \tag{11}$$

*Proof.* This follows from Equations (5), (6), (7), and (10). □

The Fibonacci sequence is obtained with $f = 2$ and $d = \infty$.

Using $f = 2$ and $d = 3$, one obtains the Padovan sequence. Indeed, for the Padovan sequence we have $F_n = F_{n-2} + F_{n-3}$ and also that $F_{n-1} = F_{n-3} + F_{n-4}$, from which one has $F_{n-3} = F_{n-1} - F_{n-4}$, and thus we have that $F_n = F_{n-2} + F_{n-3} = F_{n-1} + F_{n-2} - F_{n-4} = F_{n-1} + F_{n-f} - F_{n-d-1}$.

The $k$-step Fibonacci sequence corresponds to the case $f = 1$ and $d = k$. Indeed, for the $k$-step Fibonacci numbers, we have that

$$F_n = \sum_{i=1}^{k} F_{n-i}.$$

From

$$F_{n-1} = \sum_{i=2}^{k+1} F_{n-i} = F_{n-2} + \sum_{i=3}^{k+1} F_{n-i}$$

we have that

$$F_{n-2} = F_{n-1} - \sum_{i=3}^{k+1} F_{n-i}$$

and thus

$$\begin{aligned}
F_n &= F_{n-1} + F_{n-2} + \sum_{i=3}^{k} F_{n-i} \\
&= F_{n-1} + F_{n-1} - \sum_{i=3}^{k+1} F_{n-i} + \sum_{i=3}^{k} F_{n-i} \\
&= F_{n-1} + F_{n-1} - F_{n-k-1} \\
&= F_{n-1} + F_{n-f} - F_{n-d-1} \qquad \text{(since } f = 1 \text{ and } k = d\text{)}.
\end{aligned}$$

We also notice that, beside the case $f = 2$ and $d = \infty$, the original Fibonacci sequence is produced also by the case $f = 1$ and $d = 2$, although with a missing first term. Indeed for $f = 1$ and $d = 2$, we have that $F_1 = 1$, $F_2 = 2$ and $F_3 = 3$ (initial condition and cases 2 and 3), and then for any $n \geq d + 2 = 4$ we have that

$$\begin{aligned}
F_n &= F_{n-1} + F_{n-1} - F_{n-3} \\
&\qquad \text{(since } F_{n-1} = F_{n-2} + F_{n-2} - F_{n-4} \text{ we have)} \\
&= F_{n-1} + F_{n-2} + F_{n-2} - F_{n-3} - F_{n-4} \\
&\qquad \text{(since } F_{n-2} = F_{n-3} + F_{n-3} - F_{n-5} \text{ we have)} \\
&= F_{n-1} + F_{n-2} + F_{n-3} - F_{n-4} - F_{n-5}.
\end{aligned}$$

By iterating this reasoning we have that

$$\begin{aligned} F_n &= F_{n-1} + F_{n-2} + F_{n-4} - F_{n-5} - F_{n-6} \\ &= F_{n-1} + F_{n-2} + F_{n-5} - F_{n-6} - F_{n-7} \\ &= F_{n-1} + F_{n-2} + F_{n-6} - F_{n-7} - F_{n-8} \\ &= \ldots \\ &= F_{n-1} + F_{n-2} + F_3 - F_2 - F_1 \\ &= F_{n-1} + F_{n-2} + 3 - 2 - 1 \\ &= F_{n-1} + F_{n-2}. \end{aligned}$$

Finally we show that (2) is indeed equivalent to (11). From (2) we have

$$F_n = F_{n-f} + F_{n-f-1} + \cdots + F_{n-d}$$

and also

$$F_{n-1} = F_{n-f-1} + \cdots + F_{n-d} + F_{n-d-1}$$

from which

$$F_{n-f-1} + \cdots + F_{n-d} = F_{n-1} - F_{n-d-1}.$$

Thus we have

$$F_n = F_{n-f} + F_{n-f-1} + \cdots + F_{n-d} = F_{n-f} + F_{n-1} - F_{n-d-1}.$$

Table 4.1 summarizes the correspondence of Equation (11) with known sequences from the On-Line Encyclopedia of Integer Sequences (OEIS) [12].

# 5 Conclusions

We have given a simple recurrence relation, proved with a straightforward combinatorial argument, for the number of rabbits in the generalized Fibonacci problem, in which rabbits become fertile after an arbitrary number of generations and they also, at some point, die. The recurrence relation generalizes both the original Fibonacci sequence and other Fibonacci-related sequences, such as the Padovan sequence, the Tribonacci, Tetranacci, and alike sequences. Although the problem of counting rabbits for the dying rabbits problem has been solved in previous studies, the recurrence relation proposed in this paper has not been explicitly given previously. Moreover, the proof that we have provided in this paper is simple and elementary.

Table 4.1: Correspondence with other Fibonacci-like sequences.

| $f$ | $d$ | OEIS code | Name |
|---|---|---|---|
| 2 | $\infty$ | A000045 | Fibonacci |
| 1 | 2 | A000045 | Fibonacci |
| 1 | 3 | A000073 | Tribonacci |
| 1 | 4 | A000078 | Tetranacci |
| 1 | 5 | A001591 | Pentanacci |
| 1 | $k$ | - | $k$-step Fibonacci |
| 2 | 3 | A000931 | Padovan |
| 2 | 4 | A000930 | - |
| 2 | 5 | A072465 | - |
| 2 | 6 | A060961 | - |
| 2 | 8 | A117760 | - |
| 2 | 12 | A000044 | - |
| 3 | 4 | A079398 | - |
| 3 | 5 | A017818 | - |
| 3 | 6 | A003269 | - |
| 4 | 5 | A103372 | - |
| 4 | 7 | A017829 | - |
| 4 | 8 | A003520 | - |
| 4 | 10 | A160333 | - |

The dying rabbits problem was posed by Brother U. Alfred in the first issue of the Fibonacci Quarterly [1], probably as what the author expected to be an easy counting problem. Here is a verbatim quote from a subsequent paper [2] by Brother U. Alfred:

> Originally, it was thought that the rabbits removed would constitute a sequence which could be readily identified with an expression involving Fibonacci numbers. But after several attempts by a number of people it appeared that it would be difficult to arrive at an answer by straightforward intuition.

In this paper we have shown that, after all, Brother U. Alfred was right in considering the dying rabbits problem an easy counting problem.

## Acknowledgments

We would like to thank Paolo D'Arco, Marialaura Noce, Angel Perez Del Pozo, Douglas R. Stinson and Ugo Vaccaro for reading early versions of this paper and providing useful feedback. We also would like to thank an anonymous referee for helpful comments.

## References


[1] Brother U. Alfred, Exploring Fibonacci numbers, *Fibonacci Quart.*, **1(1)** (1963), 57–63.

[2] Brother U. Alfred, Dying rabbit problem revived, *Fibonacci Quart.*, **1(4)** (1963), 53–56.

[3] R. M. Capocelli and P. Cull, *Generalized Fibonacci numbers are rounded powers*, in "Applications of Fibonacci numbers, vol. 3", G. E. Bergum, A. N. Philippou and A. F. Horadam, eds., Springer, Dordrecht, (1990), 57–62.

[4] J. H. E. Cohn, Letter to the editor, *Fibonacci Quart.*, **2(2)** (1964), p. 108.

[5] V. E. Hoggatt, Generalized rabbits for generalized Fibonacci numbers. *Fibonacci Quart.*, **7(5)** (1969), 482–487.

[6] V. E. Hoggatt and D. A. Lind, The dying rabbit problem, *Fibonacci Quart.*, **7(5)** (1969), 482–487.

[7] T. Koshy, *Fibonacci and Lucas numbers with applications*, John Wiley and Sons, NY, 2001.

[8] M. D. Miller, On generalized Fibonacci numbers, *Amer. Math. Monthly*, **78(10)** (1971), 1108–1109.

[9] T. Noe, T. Piezas III and E. Weisstein, Webpage: Fibonacci n-step number, https://mathworld.wolfram.com/Fibonaccin-StepNumber.html, accessed: 2024-04-06.

[10] S. Vajda, *Fibonacci and Lucas numbers, and the golden section: Theory and applications*, John Wiley and Sons, NY, 2001.

[11] A. M. Oller-Marcén, The dying rabbit problem revisited, *Integers*, **9** (2009), 129–138.

[12] The On-Line Encyclopedia of Integer Sequences, https://oeis.org/, accessed: 2024-04-06.


ROBERTO DE PRISCO
DEPARTMENT OF COMPUTER SCIENCE, UNIVERSITY OF SALERNO, ITALY
robdep@unisa.it

# Appendix A Program simulation

In this section we provide the code of a Java program that simulates the growth of the population of rabbits.

```java
import java.util.*;
public class CountFibRabbits {
  static Set<Rabbit> population = new HashSet();
  public static void main(String[] args) {
     int FERTILE_AT = 3; // >= 1
     int DIES_AT = 9; // >= 0, 0 means rabbits never die
     int GENERATIONS = 35;
     //Initial population, one (couple of) rabbit(s)
     Rabbit r = new Rabbit(DIES_AT,FERTILE_AT);
     population.add(r);
     String allSizes = "Sequence:\n";
     // Evolution
     for (int n=1; n<=GENERATIONS; n++) {
         System.out.printf("n=%d  Fn=%d",n,population.size());
         // Compute and print ages only if DIES_AT>0
         if (DIES_AT>0) {
             int[] ages = new int[DIES_AT+1];
             for (int x=0; x<=DIES_AT; x++) ages[x]=0;
             for (Rabbit myr: population)ages[myr.getAge()]++;
             System.out.print(" ages=");
             for (int x=1; x<=DIES_AT; x++)
                  System.out.printf(" %d",ages[x]);
         }
         System.out.println("");
         allSizes = allSizes + population.size() + " ";
         Set<Rabbit> next = new HashSet();
         for (Rabbit rabbit : population) {
            //First increase age
            boolean alive = rabbit.increaseAge();
            //so that if rabbit is fertile, gives birth
            Rabbit child = rabbit.reproduce();
            if (child != null) next.add(child);
            //if still alive, keep it in the population
            if (alive) next.add(rabbit);
         }
         population = next;
      }
   System.out.println(allSizes);
   }
}
```

```
public class Rabbit {
    private int age;
    private int dies_at; //if 0, never dies
    private int fertile_at;
    public Rabbit(int dies_at, int fertile_at) {
        this.fertile_at = fertile_at;
        this.dies_at = dies_at;
        age=1;
    }
    public boolean increaseAge() {
        age++;
        if (dies_at < 1 || age <= dies_at) return true;
        return false;
    }
    public Rabbit reproduce() {
        Rabbit child = null;
        if (age>fertile_at) {
            child = new Rabbit(dies_at, fertile_at);
        }
        return child;
    }
    public int getAge() {
        return age;
    }
    public int getDiesAt() {
        return dies_at;
    }
}
```

The class `CountFibRabbits` contains the main program[4] that simulates the evolution by exploiting the class `Rabbit` that implements the behavior of the rabbits. The program computes the number of rabbits at each generation $n$, for small values of $n$, that is, for values of $n$ that do not cause memory problems (out of memory or overflow) to the machine running the program.

[4]To run the program in an environment like Eclipse, create a new Java project with name CountFibRabbits, create a unique package name and within the package create the two classes: the main one named CountFibRabbits and the auxiliary one named Rabbit and copy the code.

# Appendix B An example

Table B.1 shows the values of $F_n$ and $F_n^x$, for the specific case of $f = 3$ and $d = 9$ and for $n$ up to 35, obtained with the simulation program provided in Appendix A.

Table B.1: Counting table for $f = 3$ and $d = 9$.

| $n$ | $F_n$ | $F_n^1$ | $F_n^2$ | $F_n^3$ | $F_n^4$ | $F_n^5$ | $F_n^6$ | $F_n^7$ | $F_n^8$ | $F_n^9$ |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 3 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 4 | 2 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 5 | 3 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 6 | 4 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 |
| 7 | 6 | 2 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 0 |
| 8 | 9 | 3 | 2 | 1 | 1 | 1 | 0 | 0 | 1 | 0 |
| 9 | 13 | 4 | 3 | 2 | 1 | 1 | 1 | 0 | 0 | 1 |
| 10 | 18 | 6 | 4 | 3 | 2 | 1 | 1 | 1 | 0 | 0 |
| 11 | 26 | 8 | 6 | 4 | 3 | 2 | 1 | 1 | 1 | 0 |
| 12 | 38 | 12 | 8 | 6 | 4 | 3 | 2 | 1 | 1 | 1 |
| 13 | 55 | 18 | 12 | 8 | 6 | 4 | 3 | 2 | 1 | 1 |
| 14 | 79 | 25 | 18 | 12 | 8 | 6 | 4 | 3 | 2 | 1 |
| 15 | 114 | 36 | 25 | 18 | 12 | 8 | 6 | 4 | 3 | 2 |
| 16 | 165 | 53 | 36 | 25 | 18 | 12 | 8 | 6 | 4 | 3 |
| 17 | 238 | 76 | 53 | 36 | 25 | 18 | 12 | 8 | 6 | 4 |
| 18 | 343 | 109 | 76 | 53 | 36 | 25 | 18 | 12 | 8 | 6 |
| 19 | 495 | 158 | 109 | 76 | 53 | 36 | 25 | 18 | 12 | 8 |
| 20 | 715 | 228 | 158 | 109 | 76 | 53 | 36 | 25 | 18 | 12 |
| 21 | 1032 | 329 | 228 | 158 | 109 | 76 | 53 | 36 | 25 | 18 |
| 22 | 1489 | 475 | 329 | 228 | 158 | 109 | 76 | 53 | 36 | 25 |
| 23 | 2149 | 685 | 475 | 329 | 228 | 158 | 109 | 76 | 53 | 36 |
| 24 | 3102 | 989 | 685 | 475 | 329 | 228 | 158 | 109 | 76 | 53 |
| 25 | 4477 | 1428 | 989 | 685 | 475 | 329 | 228 | 158 | 109 | 76 |
| 26 | 6461 | 2060 | 1428 | 989 | 685 | 475 | 329 | 228 | 158 | 109 |
| 27 | 9325 | 2973 | 2060 | 1428 | 989 | 685 | 475 | 329 | 228 | 158 |
| 28 | 13459 | 4292 | 2973 | 2060 | 1428 | 989 | 685 | 475 | 329 | 228 |
| 29 | 19425 | 6194 | 4292 | 2973 | 2060 | 1428 | 989 | 685 | 475 | 329 |
| 30 | 28035 | 8939 | 6194 | 4292 | 2973 | 2060 | 1428 | 989 | 685 | 475 |
| 31 | 40462 | 12902 | 8939 | 6194 | 4292 | 2973 | 2060 | 1428 | 989 | 685 |
| 32 | 58398 | 18621 | 12902 | 8939 | 6194 | 4292 | 2973 | 2060 | 1428 | 989 |
| 33 | 84284 | 26875 | 18621 | 12902 | 8939 | 6194 | 4292 | 2973 | 2060 | 1428 |
| 34 | 121644 | 38788 | 26875 | 18621 | 12902 | 8939 | 6194 | 4292 | 2973 | 2060 |
| 35 | 175565 | 55981 | 38788 | 26875 | 18621 | 12902 | 8939 | 6194 | 4292 | 2973 |