1 Fast Fibonacci

Proposition 1.
$F(2n) = F(n)^2 + F(n-1)^2$
$F(2n+1) = (2.F(n-1) + F(n)).F(n)$

The proof is by induction.

- The **basis case** (for $n = 1$) is true since:

  $F(2) = F(1)^2 + F(0)^2 = 2$

  $F(3) = (2.F(0) + F(1)).(F(1)) = 3$

- **Induction step:** Let assume the property holds at rank n for both $F(2n)$ and $F(2n+1)$ and compute $F(2(n+1))$:

  Apply the definition of Fibonacci numbers:

  $F(2n+2) = F(2n+1) + F(2n)$

  Replace both terms by the recurrence hypothesis:

  $= F(n)^2 + F(n-1)^2 + (2.F(n-1) + F(n)).F(n)$

  $= F(n)^2 + F(n-1)^2 + 2.(F(n).F(n-1)) + F(n)^2$

  $= (F(n) + F(n-1))^2 + F(n)^2$

  We obtain the result by applying again the definition of Fibonacci numbers of order $n+1$:

  $F(2(n+1)) = F(n+1)^2 + F(n)^2$

  The second part of the proposition is obtained by applying the definition of Fibonacci numbers:

  $F(2(n+1)+1) = F(2(n+1)) + F(2n+1)$

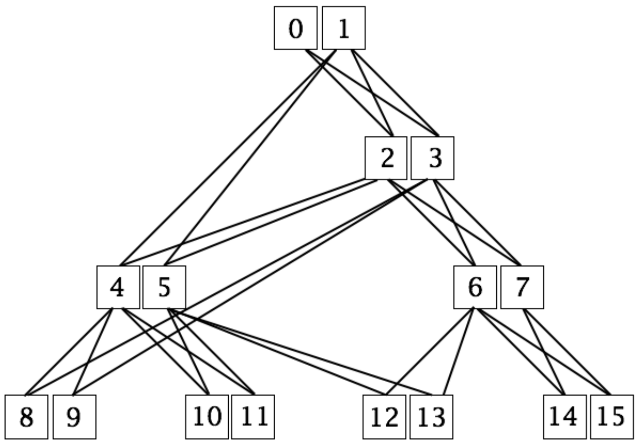
  and replace both terms by their expressions respectively :

  $= F(n+1)^2 + F(n)^2 + (2.F(n-1) + F(n)).F(n)$

  $= F(n+1)^2 + 2.(F(n-1) + F(n)).F(n)$

  $= F(n+1)^2 + 2.F(n+1).F(n)$

We provide below a pictoral argument to show why this decomposition is particularly useful while *computing* Fibonacci numbers:
We compute any $F(n)$ in $log_2(n)$ steps.



2 Proposition of Lucas' number using Fibonacci

Proposition 2.
$F(n+1) = \frac{1}{2}(F(0).L(n) + F(n).L(0))$

The proof comes from direct arithmetic manipulations:
$2.F(n+1) = F(n+1) + F(n+1) = F(n+1) + F(n) + F(n-1)$
$= L(n) + F(n)$
$= F(0).L(n) + F(n).L(0)$

The previous property can be extended for any $m > 1$ as follows:

Proposition 3.
$2.F(n+m) = F(m-1).L(n) + F(n).L(m-1)$

The proof is by recurrence on m considering any fixed n.

- The **basis case** (for $m = 1$) is given by the previous proposition.
- **Induction step:** Assume the property holds at rank $m > 1$ and consider $F(n+m+1)$:

  Apply the definition of Fibonacci numbers:

  $F(n+m+1) = F(n+m) + F(n+m-1)$

  Replace both terms by the recurrence hypothesis:

  $= \frac{1}{2}(F(m-1).L(n)+F(n).L(m-1))+\frac{1}{2}(F(m-2).L(n)+F(n).L(m-2))$
  $= \frac{1}{2}\left((F(m-1) + F(m-2)).L(n) + F(n).(L(m-1) + L(m-2))\right)$
  $= \frac{1}{2}\left(F(m).L(n) + F(n).(L(m)\right)$

3 Fibonacci system number

Using the combinatorial characterization of $F(n)$, which corresponds to the number of non consecutive bits equal to 1 in a vector of dimension n, we study here how Fibonacci numbers can be used for representing integers.

Let us first introduce a notation: $j \gg k$ iff $j \geq k+2$.

We will first prove the *Zeckendorf's theorem* which states that every positive integer n has a unique representation of the form:

$n = F_{k_1} + F_{k_2} + ... + F_{k_r}$ where $k_1 \gg k_2 \gg ... \gg k_r$ and $k_r \geq 2$.

Here, we assume that the Fibonacci sequence starts at index 1 and not 0, moreover, the decompositions will never consider $F(1)$ (since $F(1) = F(2)$). For instance, the representation of 12345 turns out to be:

$12345 = 10946+987+377+34+1 = F(21)+F(16)+F(14)+F(9)+F(2)$

Figure 1 shows the decomposition of the first integers written in this system and its principle is depicted in Figure 2.

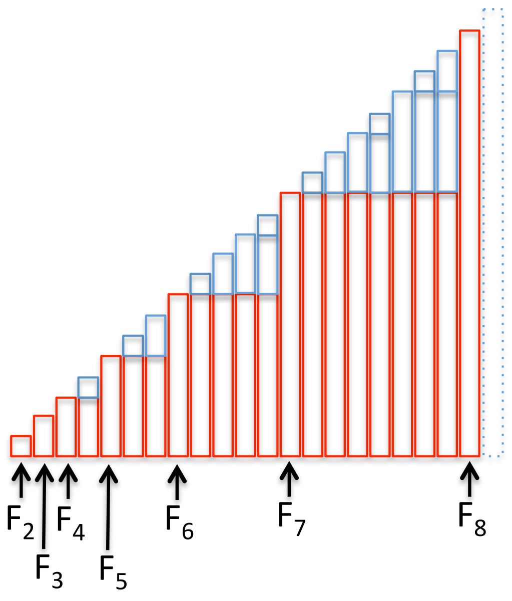


Figure 1: The first integers (on the X-axis) broken down into the Zeckendorf representation.

Proof of Zeckendorf's Theorem

The proof is done by induction on n for proving both construction and uniqueness.

- The basis is true since the decomposition is obviously unique for $n = 2$

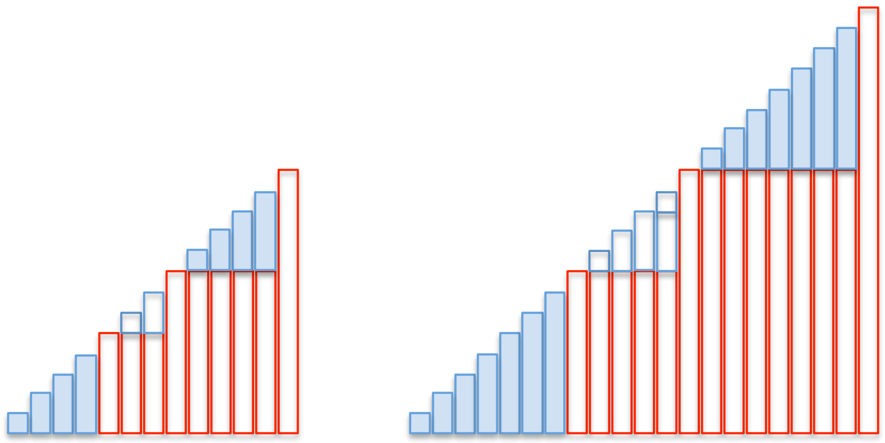


Figure 2: Principle of the construction of the Zeckendorf decomposition for the successive numbers.

(and also for $n = 3$). Notice that for $n = 4$, we have $4 = 3 + 1 = F(4) + F(2)$.

- Assume for the induction step that any integer strictly lower than $F(k)$ can be decomposed uniquely as the sum of non-consecutive Fibonacci numbers. We will prove as a consequence that an integer n in the next interval between two consecutive Fibonacci numbers $F(k) \leq n < F(k+1)$ may be decomposed.

  If $n = F(k)$ is a Fibonacci number, the decomposition is reduced to $F(k)$.

  Moreover, it is not difficult to check that it is unique.

  If $n \neq F(k)$ write $n = F(k) + N$.

  As N is strictly lower than $F(k)$, we apply the recurrence hypothesis to decompose it into non-consecutive Fibonacci numbers:

  $n = F(k) + F(k_1) + F(k_2) + \cdots + F(r)$ where $k_2 \gg ... \gg k_r \geq 2$.

  The last point to verify is that $F(k)$ and $F(k_1)$ are not consecutive ($F(k) \gg F(k_1)$), which is done by contradiction:

  Assuming k and k_1 are consecutive ($k_1 = k - 1$) leads to $n = F(k+1) + F(k_2) + \cdots + F(r)$ which contradicts $n < F(k+1)$.

Any unique system of representation is a numbering system.

The previous theorem ensures that any non-negative integer can be written as a sequence of bits b_i, in other words,

$n = (b_m b_{m-1} ... b_2)_F$ iff $n = \Sigma_{k=2}^{m} b_k F_k$.

Let us compare this system to the binary representation. For instance, the Fibonacci representation of 12345 is $(1000010100001000000 10)_F$ while $12345 = 2^{13} + 2^{12} + 2^5 + 2^4 + 2^3 + 2^0 = (1100000111001)_2$.

The binary representation is more compact.

The decomposition in the Fibonacci basis of the first integers (starting from $1 = (00001)_F$) is as follows:
$2 = (0010)_2 = F_3 = (00010)_F$
$3 = (0011)_2 = F_4 = (00100)_F$
$4 = (100)_2 = 3 + 1 = (00101)_F$
$5 = (101)_2 = F_5 = (01000)_F$
$6 = (110)_2 = 5 + 1 = (01001)_F$
$7 = (111)_2 = 5 + 2 = (01010)_F$
$8 = (1000)_2 = F_6 = (10000)_F$
$9 = (1001)_2 = (10001)_F$
$10 = (1010)_2 = (10010)_F$
$11 = (1011)_2 = (10100)_F$
$12 = (1100)_2 = (10101)_F$
$13 = (1101)_2 = F_7 = (100000)_F$
...
There is no consecutive digits equal to 1 in such representations. The proof is by contradiction and comes directly from the $\gg$ relation.

Let us sketch how to perform basic arithmetic operations within this system. We focus on the *increment* (add 1 to an integer n).

- This operation is simple if the last two digits are 00. Indeed, as there are no consecutive 1s in the coding, we simply put a 01 at the two rightmost positions.
- If the three last digits are 010, the operation gives 011, which is transformed into 100. Then, the process depends on the value of the fourth right digit. The same transformation is propagated to the left if it is a 1.
- The last case to consider is when the two last digits are 01. Adding 1 leads to 10. Then, we proceed as for the previous case if the third rightmost digit is a 1.