

Artículo

Fibonnacci y la Recursividad



David Aroesti

🕒 11 de Diciembre de 2019

La secuencia de Fibonacci es una función matemática que se define recursivamente. En el año 1202, el matemático italiano Leonardo de Pisa, también conocido como Fibonacci, encontró una fórmula para cuantificar el crecimiento que ciertas poblaciones experimentan.

Imagina que una pareja de conejos nace, un macho y una hembra, y luego son liberados. Imagina, también, que los conejos se pueden reproducir hasta la edad de un mes y que tienen un periodo de gestación también de un mes. Por último imagina que estos conejos nunca mueren y que la hembra siempre es capaz de producir una nueva pareja (un macho y una hembra). ¿Cuántos conejos existirán al final de seis meses?

Una forma de visualizar este crecimiento es mirándolo de forma tabular:

Mes	Hembras
0	1
1	1
2	2
3	3
4	5
5	8
6	13

Un punto importante a considerar es que para el mes $n > 1$, $hembras(n) = hembras(n - 1) + hembras(n - 2)$.

Como podemos ver, tenemos una definición distinta a la de factorial que vimos anteriormente. En específico, tenemos dos casos base (0 y 1) y tenemos dos llamadas recursivas ($hembras(n - 1) + hembras(n - 2)$).

Podemos crear una solución recursiva de manera sencilla:

```
def fibonacci(n):  
    if n == 0 or n == 1:  
        return 1  
  
    return fibonacci(n - 1) + fibonacci(n - 2)
```

Aunque la definición es muy sencilla, es también bastante ineficiente. En los siguientes cursos de la serie de pensamiento computacional veremos como calcular exactamente la eficiencia de este algoritmo y cómo optimizarlo. De mientras, platícanos si conoces alguna otra definición recursiva.

>

15

16

17

18

19

20

21

22

23

24