

| Complejidad    | n=20        | n=40        | n=60        |
|----------------|-------------|-------------|-------------|
| n              | 0.00002 seg | 0.00004 seg | 0.00006 seg |
| n <sup>2</sup> | 0.0004 seg  | 0.0016 seg  | 0.0036 seg  |
| n <sup>3</sup> | 0.008 seg   | 0.064 seg   |             |
| 2 <sup>n</sup> | 1 seg       | 12.7 días   | 366 siglos  |

Diagram illustrating the growth of execution time for different complexity classes as the input size  $n$  increases from 20 to 40 to 60.

**Complexity Classes and Execution Times:**

- n:** 0.00002 seg (n=20) → 0.00004 seg (n=40) → 0.00006 seg (n=60)
- n<sup>2</sup>:** 0.0004 seg (n=20) → 0.0016 seg (n=40) → 0.0036 seg (n=60)
- n<sup>3</sup>:** 0.008 seg (n=20) → 0.064 seg (n=40)
- 2<sup>n</sup>:** 1 seg (n=20) → 12.7 días (n=40) → 366 siglos (n=60)

**Growth Factors (Arrows):**

- n to n:** x 2 (orange arrow)
- n to n:** x 20 (blue arrow, labeled  $n^2/n$  veces más)
- n<sup>2</sup> to n<sup>2</sup>:** x 2 (orange arrow)
- n<sup>2</sup> to n<sup>2</sup>:** x 20 (blue arrow, labeled  $n^2/n$  veces más)
- n<sup>2</sup> to n<sup>2</sup>:** x 2<sup>2</sup> (yellow arrow)
- n<sup>2</sup> to n<sup>2</sup>:** x 3<sup>2</sup> (yellow arrow)
- n<sup>3</sup> to n<sup>3</sup>:** x 400 (red arrow, labeled  $n^3/n$  veces más)
- n<sup>3</sup> to n<sup>3</sup>:** x 2<sup>3</sup> (orange arrow)
- 2<sup>n</sup> to 2<sup>n</sup>:** x 2<sup>20</sup> (purple arrow)
- 2<sup>n</sup> to 2<sup>n</sup>:** x 2<sup>40</sup> (blue arrow)
- 2<sup>n</sup> to 2<sup>n</sup>:** x 52428 (green arrow, labeled  $2^n/n$  veces más)

Para movernos hacia abajo, lo que hay que ver es cuántas veces más de tiempo vamos a tardar, por lo que habrá que dividir lo que tardaremos con la nueva complejidad entre la complejidad de la que partamos. Por ejemplo, partiendo de  $n=20$  y complejidad  $n$  donde tardamos

0.00002s, si queremos calcular el tiempo con complejidad  $n^2$ , habrá que multiplicar por  $\frac{n^2}{n}$ , es decir, por 20.

Para movernos hacia la derecha en la tabla (partiendo de  $n=20$ ), el número de elementos se está multiplicando por dos ( $n=40$ ) y por tres ( $n=60$ ). Para saber por qué hay que multiplicar el tiempo para cada caso con  $n=20$  y obtener los otros, simplemente hay que sustituir el nuevo  $n$  por  $2n$  o  $3n$  y realizar el cociente. Por ejemplo:

- Con  $n^3$  y  $n=20$ , tenemos que el tiempo es 0.008 seg. Para conseguir el tiempo con  $n=40$ , es decir, con el doble de elementos que al principio, hay que multiplicar por  $\frac{(2n)^3}{n^3}$ , es decir, multiplicar por  $2^3$ .
- Con  $2^n$  y  $n=20$ , tenemos que el tiempo es 1 seg. Para conseguir el tiempo con  $n=40$  (el doble de elementos con respecto al  $n$  inicial, que era  $n=20$ ), hay que multiplicar por  $\frac{2^{2n}}{2^n}$ , es decir, por  $2^{20}$ . Igualmente, para conseguir el tiempo con  $n=60$ , habrá que multiplicar por  $\frac{2^{3n}}{2^n}$ , es decir, por  $2^{40}$ .

NOTA: En la tabla hay redondeos y que hacen que los errores se propaguen, pero viene bien para estimar el tiempo. Por ejemplo, en complejidad  $2^n$  con  $n=20$  el tiempo no es 1 segundo, sino 1,048576s (concretamente, el resultado de hacer  $0.00002 * 2^{20}/20$ ). De todas formas, el tiempo real no será exactamente el calculado, porque también influye el uso de memoria, recolección de basura, etc, pero es una buena estimación.