Eficiencia

(Transparencia 23)

5 diferentes algoritmos para resolver un mismo problema. La siguiente tabla establece el **tamaño máximo** de datos que han podido ser procesados:

Sabiendo un elemento de la tabla se puede calcular el resto. Por ejemplo, si sabemos que con complejidad n en un segundo podemos resolver un problema, por ejemplo de ordenación de un vector de tamaño 1000, entonces:

Algoritmo	Complejidad	1 segundo	1 minuto	1 hora
A^1	n	1000 = 10^3 elm	60 x 10^3 elm	60^2 x 10^3 elm
A^2	nlogn	140	4893	2 x 10^5
A^3	n^2	31	244	1897
A^4	n^3	10	39	153
A^5	2^n	9-10	15-16	21

- **Forma 1: si la complejidad es n^2 y conocemos la complejidad en n:**

1 segundo: si el algoritmo de **complejidad n** procesa **10^3 elementos en 1 segundo**, ¿cuántos podrá procesar el de complejidad n^2?

$$n^2 = 10^3$$
$$\boldsymbol{n = \sqrt{10^3}}$$
$$\boldsymbol{n = 10\ x\ \sqrt{10}}$$

$$\boldsymbol{n = 31,62}$$

1 minuto: si el algoritmo de **complejidad n** procesa **60 x 10^3 en 1 minuto**, ¿cuántos el de complejidad n^2?

$$n^2 = 60\ x\ 10^3$$
$$\boldsymbol{n = \sqrt{60\ x\ 10^3}}$$
$$\boldsymbol{n = 10\ x\ \sqrt{60\ x\ 10}}$$

$$\boldsymbol{n = 244,95}$$

1 hora: si el algoritmo de **complejidad n** procesa **3600 x 10^3 en 1 hora**, ¿cuántos el de complejidad n^2?

$$n^2 = 3600\ x\ 10^3$$
$$\boldsymbol{n = \sqrt{60^2 x\ 10^3}}$$
$$\boldsymbol{n = 10\ x\ \sqrt{60^2\ x\ 10}}$$

$$n = 1897,37$$

- **Forma 2: si, por el contrario, conocemos el tamaño de elementos procesados para un algoritmo con complejidad n^2 para 1 segundo, valor 31.**

1 minuto**: ¿cuál sería el tamaño en 1 minuto?**

1 seg → 31^2
60 seg → ¿n^2?

Linealmente sería: 60 seg x 31^2 = 57660 operaciones que se realizan.

$$n^2 = 31^2 \ x \ 60$$
$$n = \sqrt{57660}$$
$$n = \sqrt{31^2 \ x \ 60}$$
$$\boldsymbol{n = \ 31 \ x \sqrt{60}}$$
$$n = 240{,}12$$

es decir, aprox. 240 elementos son procesados.

Nota: en los apuntes pone: 244 al hacer la $\sqrt{60 \ x \ 10^3}$ (suponiendo que conocemos este dato en complejidad n)

1 hora**: ¿ cuál sería el tamaño en 1 hora?**
1 seg → 31^2
3600 = 60 x 60→ ¿n^2?

Linealmente sería: 60^2 x 31^2 = 3459600 operaciones que se realizan.

$$n^2 = 31^2 \ x \ 60^2$$
$$n = \sqrt{3459600}$$
$$n = \sqrt{31^2 \ x \ 60^2}$$
$$n = \ 60 \ x \ 31$$

O como en la expresión anterior:

$$\boldsymbol{n = 31\sqrt{\ 60^2}}$$
$$n = 1860$$

es decir, aprox. 1860

Nota: en los apuntes pone: 1897 al hacer la $\sqrt{60^2 \ x \ 10^3}$ (suponiendo que conocemos este dato en complejidad n)

- **Forma 1: si la complejidad es n^3 y conocemos la complejidad en n:**

1 segundo: si el algoritmo de **complejidad n** procesa **10^3 en 1 segundo**, ¿cuántos podrá procesar el de complejidad n^3?

$$n^3 = 10^3$$
$$n = \sqrt[3]{10^3}$$
$$\boldsymbol{n = 10}$$

1 minuto: si el algoritmo de **complejidad n** procesa **60 x 10^3 en 1 minuto**, ¿cuántos el de complejidad n^3?

$$n^3 = 60\ x\ 10^3$$
$$n = \sqrt[3]{60\ x\ 10^3}$$
$$\boldsymbol{n = 10\ x\ \sqrt[3]{60}}$$
$$n = 39{,}15$$

1 hora: si el algoritmo de **complejidad n** procesa **3600 x 10^3 en 1 hora**, ¿cuántos el de complejidad n^3?

$$n^3 = 3600\ x\ 10^3$$
$$n = \sqrt[3]{60^2 x\ 10^3}$$
$$\boldsymbol{n = 10\ \ x\ \sqrt[3]{60^2}}$$
$$n = 153$$

- **Forma 2: si, por el contrario, conocemos el tamaño de elementos procesados para un algoritmo con complejidad n^3 para 1 segundo, valor 10.**

1 minuto:

1 seg → 10^3
60 seg → ¿n^3?

Linealmente sería: 60 seg x 10^3 = 60000 operaciones que se realizan.

$$n^3 = 60000$$
$$n = \sqrt[3]{60\ x\ 10^3}$$
$$\boldsymbol{n = 10\ x\ \sqrt[3]{60}}$$
$$n = 39$$

1 hora:

1 seg → 10^3
3600 seg = 60^2→ ¿n^3?

- Linealmente sería: 60 x 60 x 10 x 10 x 10 = 3600000 elementos que se pueden recorrer. ¿Pero en cuantas filas de n x n?

$$n^3 = 3600000$$
$$n = \sqrt[3]{60^2\ x\ 10^3}$$
$$\boldsymbol{n = 10\ x\ \sqrt[3]{60^2}}$$
$$n = 153$$

- **Forma 1: si la complejidad es 2^n y conocemos la complejidad en n:**

Si el algoritmo de **complejidad n** procesa **10^3 en 1 segundo**, ¿cuántos podrá procesar el de complejidad $2^{n?}$

$$2^n = 10^3$$
$$log2^n = log10^3$$
$$nlog2 = \log 10^3$$

$$n = \frac{log10^3}{log2}; \quad por\ propiedades\ de\ los\ log$$

$$\boldsymbol{n = \ log_2 10^3}$$

$$n = 9{,}96$$

1 minuto: si el algoritmo de **complejidad n** procesa **60 x 10^3 en 1 minuto**, ¿cuántos el de complejidad 2^n?

$$2^n = 60\ x\ 10^3$$
$$log2^n = \log(60x10^3)$$
$$nlog2 = \log(60x10^3)$$

$$n = \frac{\log\,(60x10^3)}{log2}; \quad por\ propiedades\ de\ los\ log$$

$$\boldsymbol{n = \ log_2 60x10^3}$$

$$n = 15{,}87$$

1 hora: si el algoritmo de **complejidad n** procesa **3600 x 10^3 en 1 hora**, ¿cuántos el de complejidad 2^n?

$$2^n = 60^2\ x\ 10^3$$
$$log2^n = \log(60^2x10^3)$$
$$nlog2 = \log(60^2x10^3)$$

$$n = \frac{\log\,(60^2x10^3)}{log2}; \quad por\ propiedades\ de\ los\ log$$

$$\boldsymbol{n = \ log_2 60^2x10^3}$$

$$n = 21{,}77$$

- **Forma 2: si, por el contrario, conocemos el tamaño de elementos procesados para un algoritmo con complejidad 2^n para 1 segundo, valor 10.**

1 minuto:

1 seg → $\mathbf{2^{10}}$
60 eg → ¿2^n?

Linealmente sería: 60 seg x 2^{10} = 61440 operaciones que se realizan.

$$2^{10}x\ 60 = \ 2^n$$
$$log(2^{10}x\ 60) = \log 2^n$$
$$log(2^{10}x\ 60 = nlog2$$

$$n = \frac{\log\ (2^{10}x\ 60)}{log2}; \quad por\ propiedades\ de\ los\ log$$

$$\boldsymbol{n = \ log_2(2^{10}x\ 60)}$$

$$n = 15{,}90$$

1 hora:

1 seg → $\mathbf{2^{10}}$
3600 = 60^2→ ¿2^n?

Linealmente sería: 60^2x 2^{10} = 3686400 operaciones que se realizan.

$$2^{10}x\ 60^2 = \ 2^n$$
$$log(2^{10}x\ 60^2) = \log 2^n$$
$$log(2^{10}x\ 60^2 = nlog2$$

$$n = \frac{\log\ (2^{10}x\ 60^2)}{log2}; \quad por\ propiedades\ de\ los\ log$$

$$\boldsymbol{n = \ log_2(2^{10}x\ 60^2)}$$

$$n = 21{,}81$$

(y así sucesivamente para otros órdenes de complejidad)

- **Si la complejidad es $n\log_2 n$ y conocemos la complejidad en n:**

Si el algoritmo de **complejidad n** procesa **10^3 en 1 segundo**, ¿Cuántos procesará el de complejidad $n\log_2 n$?

$$\mathrm{n}log_2\mathrm{n} = 10^3$$
$$n = 140$$