

$$\begin{aligned}
 1s &= 1,000,000 \mu s = 1 \times 10^6 \mu s \\
 1m &= 60,000,000 \mu s = 6 \times 10^7 \mu s \\
 1h &= 3,600,000,000 \mu s = 3.6 \times 10^9 \mu s \\
 1D &= 86,400,000,000 \mu s = 8.64 \times 10^{10} \mu s \\
 1M &= 2,629,800,000,000 \mu s = 2.6298 \times 10^{12} \mu s \\
 1A &= 3,155,760,000,000,000 \mu s = 3.15576 \times 10^{13} \mu s \\
 1S &= 3.1536 \times 10^{15} \mu s
 \end{aligned}$$

Axel Giuseppe Flores Aranda
CU:181218

Estructuras de Datos Avanzadas

Tarea 1

- Para cada función $f(n)$ y tiempo t determine el tamaño máximo del problema (la n) que puede resolverse en tiempo t . Suponga que el algoritmo usado para resolver el problema toma $f(n)$ microsegundos (reporte sólo el orden de magnitud si los números son demasiado grandes)

	1 Segundo	1 Minuto	1 Hora	1 Día	1 Mes	1 Año	1 Siglo
$\log_2(n)$	$2^1 \times 10^6$	$2^6 \times 10^7$	$2^{3.6} \times 10^9$	$2^{8.64} \times 10^{10}$	$2^{2.63} \times 10^{12}$	$2^{3.16} \times 10^{13}$	$2^{3.15} \times 10^{15}$
\sqrt{n}	10^{12}	3.6×10^{15}	1.3×10^{19}	7.46×10^{21}	6.92×10^{24}	9.96×10^{26}	9.95×10^{30}
N	1×10^6	6×10^7	3.6×10^9	8.64×10^{10}	2.6×10^{12}	3.1×10^{13}	3.1×10^{15}
$n \log_2(n)$	6.3×10^4	2.8×10^6	1.3×10^8	2.7×10^{10}	7.2×10^{10}	7.9×10^{11}	6.8×10^{13}
n^2	1000	7745.97	6×10^4	2.9×10^5	1.6×10^6	5.6×10^6	5.6×10^7
n^3	100	391.48	1532.62	4420.84	13803.02	31601.04	1.4×10^5
2^n	19.93	25.83	31.74	36.33	41.25	44.84	51.48
$n!$	9	11	12	13	15	16	17

$$n! = n \cdot (n-1) \cdot (n-2) \cdot \dots \cdot 2 \cdot 1$$

- Supongamos que estamos comparando el desempeño de dos algoritmos de ordenamiento. Para entradas de tamaño n , el algoritmo A toma $8n^2$ operaciones mientras que el algoritmo B toma $64n \log_2(n)$. ¿Para qué valores de n es mejor el desempeño de A?
- ¿Cuál es el valor más chico de n para el cual un algoritmo que toma $100n^2$ es más rápido que uno que toma 2^n (en la misma máquina)?
- Demuestre que $2^n = O(n^2)$

$$2) 8n^2 = 64n \log_2(n) \Rightarrow A \text{ tiene mejor desempeño en el intervalo } n \in [1.1, 43.56]$$

$$n = 8 \log_2(n)$$

$$3) 2^n = 100n^2 \Rightarrow n \in [0, 0.14]$$

$$4) 2^n = O(n^2)$$

$$2^n = cn^2$$

$$\Rightarrow c = \frac{2^n}{n^2}, n \neq 0$$