

Andres Quevedo  
181 336

## Estructuras de Datos Avanzadas

### Tarea 1

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
$\log_2(n)$	$2^{10^6}$	$2^{6.0 \times 10^7}$	$2^{3.6 \times 10^9}$	$2^{8.6 \times 10^{11}}$	$2^{2.51 \times 10^{12}}$	$2^{3.96 \times 10^{13}}$	$2^{3.96 \times 10^{14}}$
$\sqrt{n}$	$10^{12}$	$3.6 \times 10^{15}$	$1.296 \times 10^{19}$	$7.396 \times 10^{21}$	$6.65 \times 10^{24}$	$1.57 \times 10^{27}$	$1.57 \times 10^{29}$
$N$	$10^6$	$6.0 \times 10^7$	$3.6 \times 10^9$	$8.6 \times 10^{10}$	$2.58 \times 10^{12}$	$3.96 \times 10^{13}$	$3.96 \times 10^{14}$
$n \log_2(n)$							
$n^2$	$10^3$	$7.71 \times 10^3$	$6.0 \times 10^4$	$2.93 \times 10^5$	$1.60 \times 10^6$	$6.29 \times 10^6$	$1.98 \times 10^7$
$n^3$	100	391.49	1532.62	4714.0	13715.3	37085.14	73494.2
$2^n$	19.93	25.84	31.75	36.32	41.23	45.17	48.49
$n!$	10	12	13	14	16	17	18

2. 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?
3. ¿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)?
4. Demuestre que  $2^n = O(n^2)$

2.  $8n^2 < 64n \log_2 n$

$$1 < 8 \frac{1}{n} \log_2 n$$

Grafique  $64n \log_2 n - 8n^2 > 0$

y  $8n^2 < 64n \log_2 n \forall n \in (1.1, 43)$

3.  $100n^2 < 2^n$

Grafico  $100n^2 - 2^n < 0$

$\Rightarrow 100n^2 < 2^n, \forall n \in (-0.1, 0.1)$

4.