



## Introducción a las ciencias de la computación

### Tarea 3

Fecha de entrega: viernes 14 de octubre

- Suponga que un algoritmo utiliza  $t = f(n)$  pasos para resolver cierto problema, donde  $n$  es el tamaño de la entrada del problema. Además, suponga que cada paso toma un microsegundo<sup>1</sup> para ejecutarse. Así,  $f(n)$  sería el número de microsegundos que tardaría el algoritmo en ser ejecutado. Para cada función  $f(n)$  de la siguiente tabla, determine el tiempo que tardaría en terminar el algoritmo según el tamaño de entrada  $n$ .

$f(n) \backslash n$	8	16	32	64	128	256	512	1024
$7n$								
$\log_2 n$								
$\sqrt{n}$								
$n \log_2 n$								
$n^4$								
$n^6$								
$2^n$								
$n!$								
$n^n$								

- Sin necesidad de realizar algún análisis de complejidad y basado simplemente en el tiempo que se tardan, diga cuáles algoritmos de la tabla anterior considera intratables y por qué<sup>2</sup>. Para responder a esta pregunta tal vez sea útil tener en mente los siguientes datos:

- El número de protones en el universo es  $10^{80}$ .
- El número de microsegundos desde el Big Bang es  $4.73 \times 10^{23}$  (15 mil millones de años).
- La edad de la Tierra es de  $1 \times 10^{23}$  microsegundos (4.55 mil millones de años).

<sup>1</sup>1 segundo tiene  $10^6$  microsegundos.

<sup>2</sup>Aunque la forma más precisa para determinar si un problema es intratable o no es determinando si su función de complejidad  $f(n)$  está acotada por arriba o por abajo por un polinomio.