

# Actividad 1

## CComplejidad de Algoritmos

Análisis y Diseño de Algoritmos / Ingeniería Civil Informática  
Departamento Ciencias de la Computación y  
Tecnologías de la Información

Universidad del Bío-Bío

Profesores: Gilberto Gutiérrez / Carlos San Juan

Otoño 2018

1. **Cotas superior de funciones.**

Para cada una de las funciones encuentre su cota superior, es decir,  $O(g(n))$ . En cada caso debe encontrar  $c$ ,  $n_0$  y  $g(n)$ .

(a)  $t(n) = 2n + 5$

(b)  $t(n) = 50n^2 + 4n$

(c)  $t(n) = 8n^3 + 2n \lg n$

(d)  $t(n) = \frac{n(n-1)}{2} + n^2\sqrt{n}$

2. Analice el algoritmo *InsertSort* (visto en clases) y obtenga la cantidad de comparaciones, en función del tamaño de la entrada ( $n$ ), que realiza el algoritmo en el peor de los casos.
3. Suponga que el algoritmo *MergeSort* (que estudiaremos más adelante) en el peor de los casos, ordena un conjunto de objetos realizando  $t(n) = 64n \log_2 n$  comparaciones, donde  $n$  es el tamaño de la entrada. ¿Para que valores de  $n$  *InsertSort* es mejor que *MergeSort*, en el peor de los casos?. Hint. Puede graficar las funciones de *InsertSort* y *MergeSort* para ciertos valores de  $n$ .