

Análisis asintótico de algoritmos

Sergio Revilla Velasco

Resumen

El objetivo de esta práctica es estudiar la complejidad asintótica de los algoritmos de selección(selection-sort), inserción(insertion-sort), burbuja (bubble-sort), ordenación rápida (quick-sort) y ordenación por mezcla (merge-sort) por medio de un programa escrito en C++

Índice

1. Instrucciones de compilación y dependencias	2
2. Método de obtención de tiempos y gráficas	3
3. Gráficas de tiempos obtenidas	4
4. Análisis de las gráficas	8
5. Comentarios adicionales	9

1. Instrucciones de compilación y dependencias

1. Dependencias

- Windows: MinGW <http://www.mingw.org>
- Linux: GCC Compiler Collection <http://gcc.gnu.org>
- Mac Os X: Xcode Tools con utilidades de línea de comando instaladas <https://developer.apple.com/xcode/>
- GnuPlot: usado para generar las gráficas desde archivos de texto <http://www.gnuplot.info>

2. Instrucciones de compilación

- Compilar el archivo *main.cpp* contenido en la carpeta */src*
- El programa generará archivos *txt* separado por líneas, con el tamaño del problema y el tiempo empleado en la ordenación expresado en milisegundos (*ms*).
- Ejecutar desde la consola el comando *gnuplot script.gnuplot* desde el mismo directorio donde se generaron los archivos de datos.
- El script generará archivos *png* con las gráficas.

2. Método de obtención de tiempos y gráficas

1. El programa preguntará por el tamaño del problema (la longitud del array)
2. También podemos elegir el mayor entero a generar (un entero puede tener un valor máximo de 32767)
3. El programa preguntará por el salto (*gap*) entre iteraciones. En primera instancia se genera un array con números aleatorios que sirve como base para las ordenaciones. De esta manera las copias del array original se hacen del tamaño que corresponde a la iteración, en lugar de hacer copias con el tamaño original del problema y así evitar problemas de memoria para tamaños del problema muy grandes ($n > 20000$)

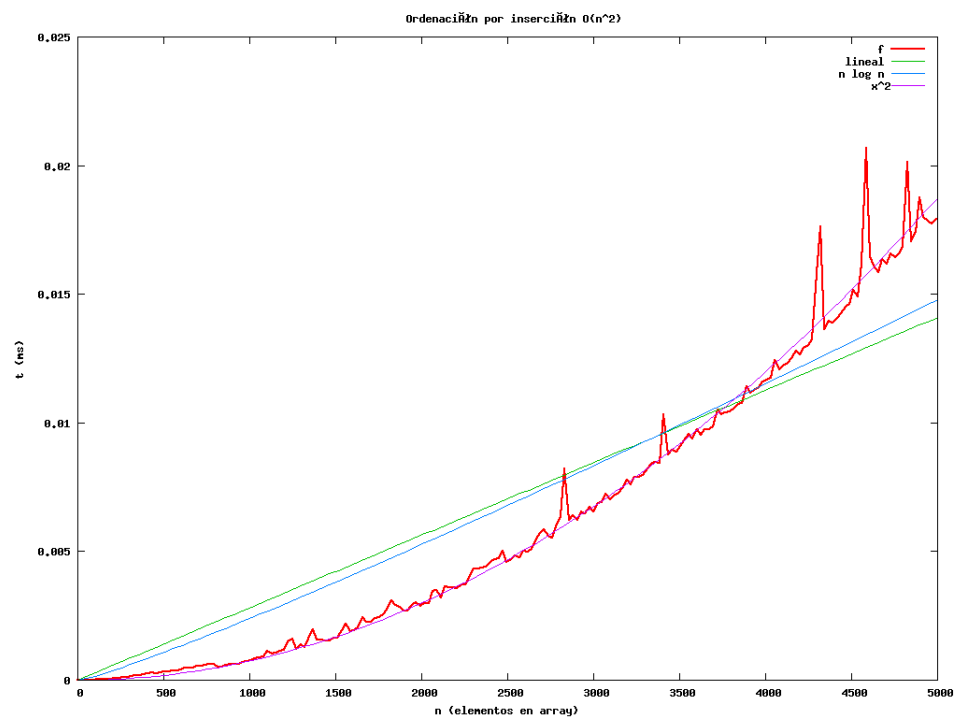


Figura 1: Ordenación por inserción

3. Gráficas de tiempos obtenidas

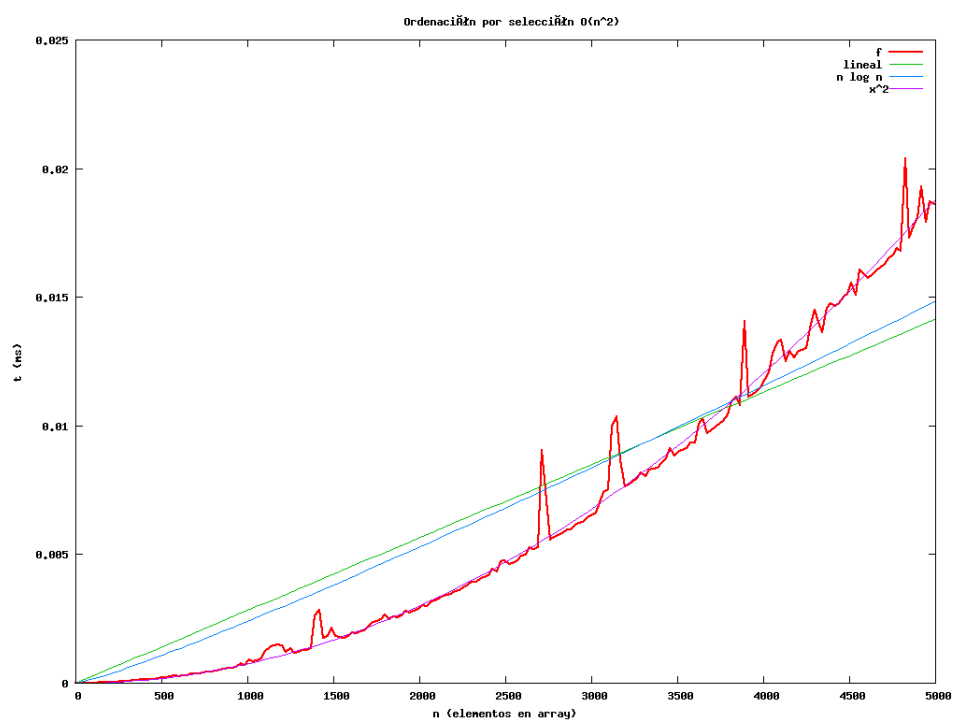


Figura 2: Ordenación por selección

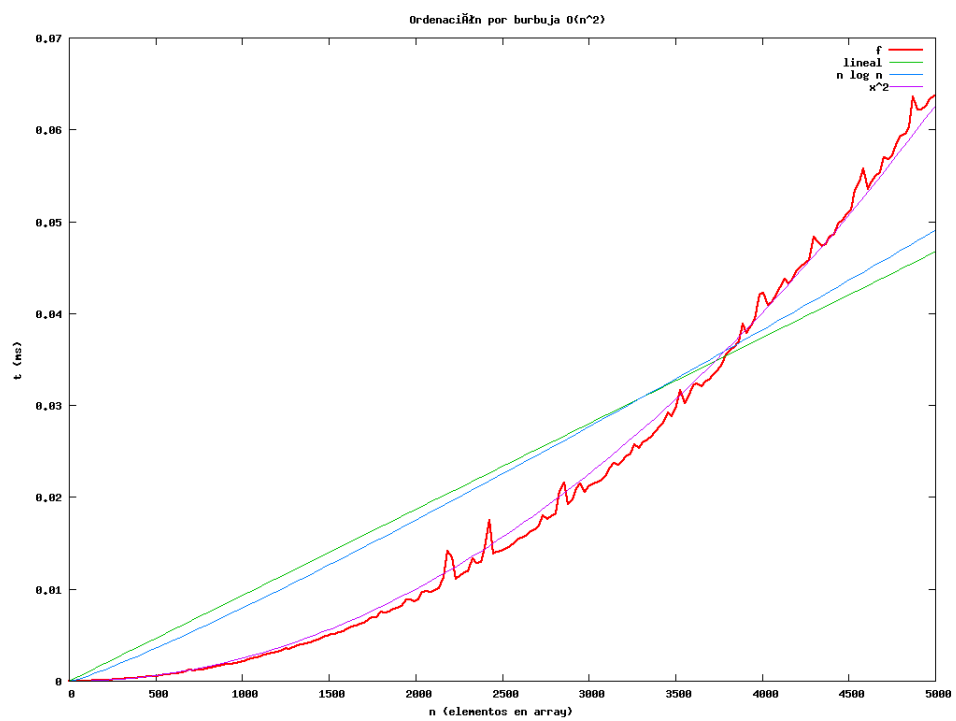


Figura 3: Ordenación por burbuja

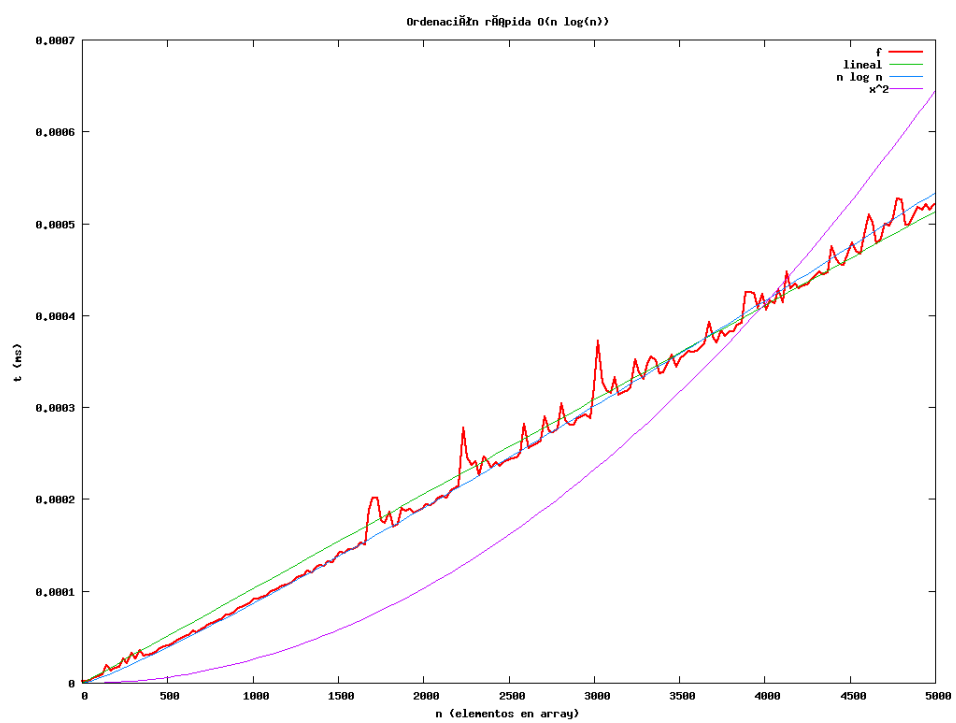


Figura 4: Ordenación rápida

4. Análisis de las gráficas

5. Comentarios adicionales