



ESCUELA UNIVERSITARIA POLITÉCNICA  
Departamento de Ciencias Politécnicas  
Grado en Ingeniería Informática

Prácticas Algoritmia  
Boletín 3. Algoritmos de clasificación

Curso 2017/2018

Profesor: Andrés Muñoz

# Boletín 3. Algoritmos de clasificación

---

## 1. Objetivo

El objetivo de este boletín es que los alumnos analicen, codifiquen y comparen algoritmos de clasificación en C.

## 2. Ejercicios

Se desea comparar los algoritmos de clasificación vistos en clase para medir experimentalmente su eficiencia. Para ello se pide implementar los algoritmos indicados en los puntos 1-4 y a continuación realizar las pruebas indicadas en el punto 5.

Algoritmos de clasificación a implementar:

**1. (0,5 puntos) Burbuja:**

- a. Código original (diapositiva 8 del Tema 3)
- b. Código mejorado (mejora a decidir por el alumno, explicar en memoria)

**2. (0,5 puntos) Selección directa:**

- a. Código original (diapositiva 14 del Tema 3)
- b. Mejorado para evitar que el pivote se intercambie por él mismo (explicar en memoria).

**3. (1,5 puntos) Método Shell** (secuencia de saltos a elegir por el alumno, explicar en memoria)

**4. (2,5 puntos) QuickSort**, en las siguientes tres variantes:

- a. Pivote: Valor en la posición del medio (proporcionado por el profesor en el fichero “Quick-Sort-Medio.txt” colgado junto a este enunciado).
- b. Pivote: Aleatorio (explicar en memoria)
- c. Pivote: Mediana utilizando el primer, último y medio elemento (explicar en memoria).

**5. (5 puntos)** Tras implementar los algoritmos, se deben ejecutar con los siguientes tres ficheros de números enteros generados aleatoriamente, con rango [0-999.999] y que pueden contener números repetidos:

- a. numeros50000.txt → Contiene 50.000 números
- b. numeros100000.txt → Contiene 100.000 números
- c. numeros200000.txt → Contiene 200.000 números

Se debe comprobar obligatoriamente que los algoritmos ordenan correctamente (de menor a mayor, o de mayor a menor) los ficheros utilizados. Se recomienda guardar los números ordenados en un nuevo fichero, ya que se necesitarán a continuación.

Para cada algoritmo se debe medir:

- i. Número de comparaciones realizadas
- ii. Número de intercambios realizados
- iii. Tiempo empleado en la ordenación (sin contar el tiempo de abrir y leer el fichero)

**Responder a las siguientes cuestiones:**

- **Comentar los resultados obtenidos, comparándolos entre ellos mediante tablas o gráficas. Indicar de manera justificada cuál es el mejor algoritmo y cuál es el peor, y si los resultados experimentales coinciden con los estudios teóricos vistos en clase.**
- **A continuación, repetir las comparaciones usando ahora los tres ficheros, pero ordenados de menor a mayor ¿Qué ocurre ahora? Compara los resultados con los obtenidos anteriormente usando los ficheros aleatorios originales.**

**Ejercicios opcionales (Cada ejercicio correcto suma 0,5 directamente a la nota final de prácticas):**

- Opción a) → Ordenación Bucket (incluir su estudio experimental en el apartado 5)
- Opción b) → Mezcla directa o natural utilizando el fichero “numeros200000.txt” (incluir su estudio experimental en el apartado 5)

### **3. Entregables y puntuación**

- Memoria que contenga las respuestas e información pedida en los ejercicios del boletín (NO INCLUIR EL CÓDIGO DE LOS PROGRAMAS). **Es especialmente importante realizar correctamente el estudio comparativo pedido en el ejercicio 5 para una buena valoración del boletín.**
- Un proyecto de Dev-C++ que incluya los ficheros .c con el código de los programas y que esté listo para ser compilado y ejecutado (NO ENTREGAD LOS FICHEROS DE CADA EJERCICIO POR SEPARADO O NO SE CORREGIRÁN LAS PRÁCTICAS).
- **Fecha de entrega: 14 de Enero 2018**
- **Nota del boletín: 10 puntos**