

UNIDAD TEMÁTICA 9 – CLASIFICACIÓN

Trabajo de Aplicación 4

EJERCICIO 1

TRABAJANDO EN SUB-EQUIPOS, SEPARADOS.

PARTE 1

Utilizando las clases JAVA provistas por la Cátedra “*TClasificador.java*” y el fragmento de código para el algoritmo de *Quicksort*, implementar completamente el mismo.

Observar que el fragmento de código provisto **contiene errores, que deberán ser identificados y reparados.**

NO SE PROVEE la función “*encuentraPivote*”, que deberá ser implementada de acuerdo a las conclusiones a que el Equipo arribe en el **TA3** - Ejercicio 1.

Se requiere:

1. Encontrar y reparar los errores en el método de *Quicksort*.
2. Agregar lo necesario para controlar el nivel de profundidad de llamada recursiva.
3. Implementar la función “*encuentraPivote*” en **más de una forma**.
4. Probar la ejecución del algoritmo de *Quicksort* con conjuntos de datos de 300 y 10,000 elementos, cada uno en tres órdenes: ascendente, descendente y aleatorio, con más de una función de pivote.
5. En las pruebas del punto 4. , emitir el nivel máximo de profundidad de llamada recursiva alcanzado, y compararlo con el ideal (logarítmico).

PARTE 2

Integrar lo desarrollado por los sub-equipos en el repositorio GIT.

Trabajo de Aplicación 4

EJERCICIO 2

Para cada uno de los algoritmos de ordenación ***Inserción Directa, Shellshort, Burbuja y Quicksort***,

1. Generar conjuntos de datos de 300, 10000 y 30000 elementos, ordenados en forma ascendente, descendente y aleatoria
2. Ejecutar la ordenación para cada conjunto con cada algoritmo
3. Una vez ejecutada la ordenación se debe verificar que el conjunto resultante ha quedado efectivamente ordenado (invocar a un método que lo verifique).
4. En cada caso (ejs: 300 elementos, ascendente; 10000 elementos, aleatorio), ¿qué algoritmo ejecuta más rápido? (utilizar una planilla electrónica para hacer las comparaciones)

Se desea analizar y realizar gráficos comparativos de las mediciones:

- 1) Dentro de cada algoritmo, 3 series para los distintos conjuntos de datos
 - a) Ascendente
 - b) Descendente
 - c) Aleatorio
- 2) A partir de los datos de la página principal, crear **gráficos de barras** comparativos entre algoritmos, para el mismo tamaño de conjunto de datos.
 - a) Para tamaño de conjunto de datos = 300 elementos (3 por cada uno de los 4 algoritmos, para conjuntos ascendente, descendente y aleatorio).
 - b) Ídem para tamaño de conjunto de datos = 10000 elementos.
 - c) Ídem para tamaño de conjunto de datos = 30000 elementos.

DESARROLLO:

PARTE 1:

Trabajando en **2 sub-equipos**, distribuir los algoritmos a medir y, colaborativamente (editando la planilla en Google Drive, Teams, etc.) , completar la planilla electrónica (en las pestañas correspondientes a cada algoritmo, NO MODIFICAR LA PESTAÑA PRINCIPAL – ésta sólo tiene referencias a los resultados de las pestañas individuales). En cada pestaña de algoritmo agregar los gráficos específicos para ese algoritmo.

PARTE 2:

Todo el Equipo: agregar los gráficos en la pestaña principal, de comparación entre algoritmos.

IMPORTANTE: NORMALIZAR DATOS Y EVENTUALMENTE UTILIZAR ESCALAS LOGARÍTMICAS EN LAS ORDENADAS

5. Responder a las preguntas presentadas en pantalla

NOTA: UTILIZAR LA PLANILLA DE CÁLCULO PROVISTA PARA REGISTRAR LOS TIEMPOS!!!

Subir las conclusiones en un archivo de texto a la tarea correspondiente.