

Estructuras de datos avanzadas

**Tarea 2**

**Comparación de algoritmos de ordenamiento de palabras: MergeSort y TrieSort**

**Diego Hernández Delgado**

Clave única: 176262

Grupo: 001

Profesor: Fernando Esponda

Fecha de entrega: 20-noviembre-2019

**Índice**

[**OBJETIVO 3**](#_Toc25574584)

[**INTRODUCCIÓN 3**](#_Toc25574585)

[**RESULTADOS 3**](#_Toc25574586)

[**ANÁLISIS DE RESULTADOS 5**](#_Toc25574587)

[**CONCLUSIÓN 6**](#_Toc25574588)

[**REFERENCIAS 6**](#_Toc25574589)

OBJETIVO

El objetivo de esta tarea es determinar empíricamente qué algoritmo entre el algoritmo MergeSort y el algoritmo TrieSort es más eficiente para ordenar un conjunto de palabras u objetos que se puedan comparar por medio de una llave (*key*).

INTRODUCCIÓN

En la presente tarea se realizó una comparación empírica de la eficiencia de los algoritmos MergeSort y TrieSort para ordenar un conjunto de elementos por sus llaves.

Para este experimento se leyó, en una aplicación de JAVA, un archivo de texto con múltiples palabras que se almacenaron en un ArrayList y, posteriormente, se almacenaron en un arreglo.

A continuación, se cronometró el tiempo que tardó cada uno de los algoritmos para ordenar las palabras del arreglo en orden lexicográfico. Este proceso se repitió 10 veces con distintas cantidades de palabras (de 10,000 a 28,000 palabras, avanzando 2,000 palabras por cada iteración). A su vez, esto último se repitió 3 veces con la intención de promediar los resultados y tener un menor margen de error.

Después de obtener todos los resultados y sus respectivos promedios, se graficaron los datos para comparar los algoritmos y obtener las conclusiones pertinentes.

RESULTADOS

|  |  |  |
| --- | --- | --- |
| Experimento 1 | | |
|
| CantPalabras | **Merge** | **Trie** |
| **(ms)** | |
| 10000 | 21 | 32 |
| 12000 | 15 | 16 |
| 14000 | 27 | 17 |
| 16000 | 76 | 14 |
| 18000 | 64 | 27 |
| 20000 | 61 | 14 |
| 22000 | 72 | 29 |
| 24000 | 73 | 12 |
| 26000 | 48 | 20 |
| 28000 | 16 | 39 |

|  |  |  |
| --- | --- | --- |
| Experimento 2 | | |
|
| CantPalabras | **Merge** | **Trie** |
| **(ms)** | |
| 10000 | 20 | 29 |
| 12000 | 20 | 18 |
| 14000 | 48 | 12 |
| 16000 | 53 | 12 |
| 18000 | 55 | 21 |
| 20000 | 55 | 11 |
| 22000 | 51 | 24 |
| 24000 | 19 | 12 |
| 26000 | 12 | 15 |
| 28000 | 11 | 32 |

|  |  |  |
| --- | --- | --- |
| Experimento 3 | | |
|
| CantPalabras | **Merge** | **Trie** |
| **(ms)** | |
| 10000 | 21 | 29 |
| 12000 | 14 | 15 |
| 14000 | 14 | 10 |
| 16000 | 19 | 14 |
| 18000 | 28 | 19 |
| 20000 | 27 | 12 |
| 22000 | 67 | 22 |
| 24000 | 22 | 14 |
| 26000 | 20 | 23 |
| 28000 | 14 | 31 |

|  |  |  |
| --- | --- | --- |
| Promedio | | |
|
| CantPalabras | **Merge** | **Trie** |
| **(ms)** | |
| 10000 | 20.6666667 | 30 |
| 12000 | 16.3333333 | 16.3333333 |
| 14000 | 29.6666667 | 13 |
| 16000 | 49.3333333 | 13.3333333 |
| 18000 | 49 | 22.3333333 |
| 20000 | 47.6666667 | 12.3333333 |
| 22000 | 63.3333333 | 25 |
| 24000 | 38 | 12.6666667 |
| 26000 | 26.6666667 | 19.3333333 |
| 28000 | 13.6666667 | 34 |

ANÁLISIS DE RESULTADOS

La gráfica nos permite visualizar que antes de las 12,000 palabras, el algoritmo de MergeSort es más eficiente que el TrieSort. Sin embargo, en el rango de 12,000 a un poco más de 26,000 palabras el algoritmo de TrieSort muestra una ventaja contundente y, posteriormente, el algoritmo MergeSort vuelve a ser más eficiente.

CONCLUSIÓN

Esta tarea ha permitido que visualicemos de una manera más cercana las diferencias, ventajas y desventajas de los algoritmos de ordenamiento MergeSort y TrieSort. Así mismo, ha permitido que desarrollemos un carácter más crítico frente a la eficiencia de los algoritmos, pues al trabajar con un gran número de datos, la diferencia en tiempo de espera y de la memoria que utiliza la computadora puede ser brutal.

REFERENCIAS

Lewis, J & Chase, J. (2006) Estructuras de datos con Java. Diseño de estructuras y algoritmos. Pearson-Addison Wesley.

Cormen, T., Leiserson C., Rivest, R., Stein, C. (2009). Introduction to Algorithms. 3a Edición. The MIT Press.

Cairó, O. y Guardati, S. (2006). Estructuras de Datos. 3era. edición. Mc. Graw Hill.