

Ejercicio 9: mapas

Un mapa es un Tipo Abstracto de Datos (TAD) diseñado para guardar y recuperar elementos de manera eficiente utilizando una clave de búsqueda única para cada elemento. En el código que se ofrece junto a este enunciado, se pueden encontrar 4 ficheros:

- `Map`: ofrece una interfaz del TAD “Mapa”.
- `AbstractMap`: ofrece una clase abstracta que implemente parte de la interfaz `Map`.
- `UnsortedArrayMap`: ofrece una implementación de la interfaz `Map` extendiendo la clase abstracta `AbstractMap`.
- `WordCount`: ofrece un conjunto de pruebas unitarias que se describirán más adelante.

Tarea 1

Se pide implementar los siguientes métodos de la clase `UnsortedArrayMap`:

- `V put(K key, V value)`
- `boolean containsKey(K key)`

La funcionalidad de estas funciones se describe en el código.

Una vez se han implementado esos métodos, el mapa se utilizará para contar la frecuencia de las diferentes palabras que forman un texto. En particular, el primer test del fichero `WordCount` (`countWordsInText()`) lee un fichero de texto (`text.txt`, en la carpeta raíz del repositorio) que contiene un fragmento de Don Quijote de la Mancha. Esta prueba utiliza el mapa implementado en `UnsortedArrayMap` para contar el número de veces que aparece cada palabra en el texto y, a continuación, imprime por pantalla la palabra más frecuente y el número de veces que esta aparece.

Tarea 2

Aunque esta primera implementación sirve a un propósito didáctico (que el estudiante se familiarice con la funcionalidad de los mapas), su eficiencia es más que cuestionable. En los próximos días veremos cómo implementar mapas de manera más eficiente con tablas hash. Para comparar el rendimiento de la clase que acabamos de implementar con el rendimiento de un `HashMap` estándar, las pruebas `timeConsumptionOfUnsortedArrayMap()` y `timeConsumptionOfHashMap()` realizan un conjunto de operaciones sobre ambas estructuras de datos y pintan por pantalla el tiempo total empleado en hacer dichas operaciones. Como se puede comprobar, la diferencia es considerable.