Memoria de la práctica 2

Búsqueda y Minería de la Información

Jorge Cifuentes, Alejandro Martín, pareja 01

Ejercicios realizados

Ejercicio 1

* 1. Motor de búsqueda orientado a términos: TermBasedVSMEngine

Hemos usado una Hashtable de pares (Integer – Double) como acumulador de los valores sumados al recorrer cada Posting List.

* 1. Motor de búsqueda orientado a documentos: DocBasedVSMEngine

Simulamos el comportamiento del Heap de postings usando una lista de iteradores y nuevamente una Hashtable (ver código)

* 1. Heap de ranking: RankingIteratorImpl

Para el Heap de ranking, el cual es un MaxHeap, hemos usado la clase ya implementada de Lucene *priorityQueue.* Para que esta cola de prioridad funcione efectivamente como un MaxHeap, le pasamos a su constructor una implementación de *lessThan* acorde. Con eso, ya tenemos el funcionamiento de MaxHeap: en el next del iterador hacemos un pop, que devuelve el máximo y lo elimina del Heap.

Cabe mencionar que esto hace que solo se pueda iterar una vez sobre él, dado que el heap se va vaciando, lo cual es el comportamiento esperado para el heap de ranking.

Ejercicio 2

2.1) SerializedRAMIndex

2.2) SerializeRAMIndexBuilder

Esta es la clase importante.