

EDA (E.T.S. de Ingeniería Informática)

Curso 2016-2017

Práctica 3 - Parte II: Uso de una Tabla de Dispersión para la generación de índices de palabras de libros

Objetivos formativos y trabajo previo



El objetivo final de esta práctica es el uso de una Tabla de Dispersión o *Tabla Hash* para implementar un *Map*. En particular se plantea la creación y consulta de un índice de palabras para una colección de libros en formato de texto.

Además se reforzarán los objetivos transversales a todas las prácticas de la asignatura y que están relacionados con la calidad de los programas desarrollados: utilización de paquetes para facilitar la organización, reutilización y mantenimiento del software, utilización de los mecanismos de herencia y genericidad que proporciona el lenguaje, elaboración de juegos de prueba para validar código y generación de documentación asociada al código desarrollado. Para aprovechar al máximo la sesión de laboratorio, antes se debe realizar una lectura comprensiva de este boletín y del código de las clases que se proporcionan a través de PoliformaT.

1. Índices de palabras para un conjunto de libros

En la actualidad existe una enorme cantidad de información disponible. El problema de tener tanta información es que no se dispone de tiempo suficiente para consultarla de manera exhaustiva. Por lo que tenemos que idear métodos eficaces para centrar la búsqueda en aquellos documentos que puedan contener la información que nos interesa. Un ejemplo muy claro de ello son los buscadores de Internet tipo Google.

Si se dispone de un repositorio de documentos de texto, en nuestro caso una biblioteca de libros, nos puede interesar un sistema para buscar términos en ella, de manera que nos guíe a las líneas de los documentos y libros que los contengan.

En esta práctica se va a utilizar una tabla hash para almacenar los índices de las palabras contenidas en toda la biblioteca. Dichos índices guardarán el libro y la línea en que aparece cada palabra de la biblioteca.

2. Las clases de la aplicación Indexador e Indice

La clase `Indexador` tiene un solo atributo, una tabla hash `Map<String, ListaConPI<Indice>>`. Las claves del serán las palabras, mientras que la información asociada a cada palabra será una `listaConPI<Indice>` que contendrá los índices para dicha palabra. Los objetos de tipo `Indice` tienen dos atributos: un `String` que contendrá el nombre del libro, y un entero que contendrá la línea donde se encuentra la palabra en dicho libro.

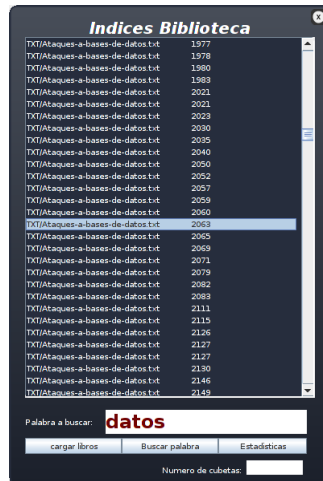


Figura 1: Captura de la ejecución de la aplicación

3. Trabajo de laboratorio

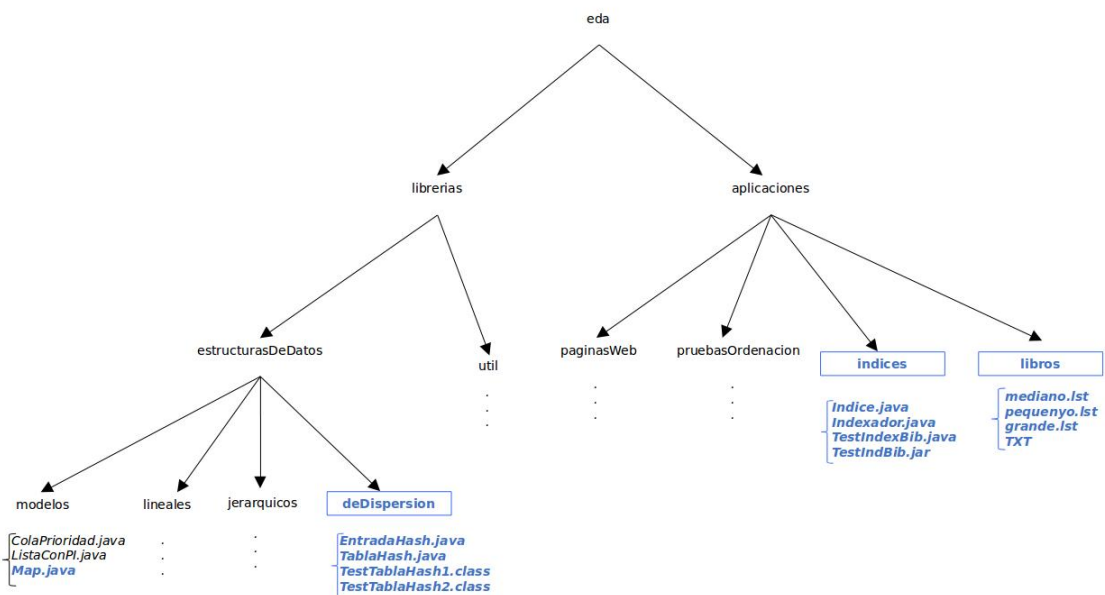


Figura 2: Captura de la ejecución de la aplicación

Actividad 1: Paquete indices

- Crear un nuevo paquete, `aplicaciones.indices`, que contendrá la aplicación destinada a la obtención, para una palabra y una colección de libros, de las líneas de los libros en las que aparece. En él añadir `testIndBib.jar` y las clases `Indexador` `Indicey` `TestIndexBib`. Nota: asegurarse que el bluej(Preferencias→librerias) tiene el path del `testIndBib.jar`.

Actividad 2 Implementar el método buscarPalabra

- Completar el método `buscarPalabra` de la clase `Indexador`, que nos devolverá los índices para la palabra deseada.

Actividad 3: uso de la aplicación `TestIndexBib`

Para finalizar la práctica se comprobará el correcto funcionamiento de la aplicación `TestIndexBib`. Para ello se deberá:

- Crear la carpeta `aplicaciones.libros`, copiar los libros (`libros.tgz`) y descomprimir el fichero (`tar xvzf libros.tgz`).
- Ejecutar el `main(String[] args)` de la clase `TestIndexBib`.

Actividad 4: uso de la aplicación `TestIndexBib`: factor de carga

- Bloquea en el método `insertar` de la clase `TablaHash` la llamada a `rehashing`.
- Ajusta en la aplicación gráfica el número de cubetas deseado.
- Carga cada colección de libros y estudia sus estadísticas.

Nótese que la primera acción que se tiene que realizar es la carga del fichero de texto que contendrá una lista de los libros a cargar.