

La programación funcional ha sido tradicionalmente considerada únicamente como un ejercicio académico. Sin embargo, los requerimientos introducidos por las últimas tendencias en tecnologías de información que involucran gigantescos volúmenes de datos distribuidos, han levantado un renovado interés en este paradigma.

El paradigma funcional es particularmente popular en aplicaciones que requieren de módulos de análisis de datos y en ambientes altamente concurrentes o distribuidos. (https://wiki.haskell.org/Haskell_in_industry)

Este proyecto pretende ilustrar una posible aplicación en el campo de la recuperación de información. El objetivo es construir un indexador / buscador de archivos locales.

Especificación

Construir dos programas en Haskell, uno para indexar un directorio del sistema de archivos local y otro que permita buscar archivos a través de palabras clave utilizando el índice generado por el primero.

- Ambas aplicaciones deben correr como ejecutables en línea de comando.
- El directorio a indexar puede tener cualquier profundidad y anchura (ej. podría ser /).
- El indexador recibirá como argumento la ruta al directorio a indexar.
- El índice debe manejarse como un trie.
- El indexador sólo procesará archivos de texto (debe proveer la lista de extensiones de archivos de texto que se soportarán).
- Para cada archivo se procesarán todas las palabras que no pertenezcan al conjunto de *stop words* (español e inglés).
- El indexador será concurrente, cada directorio debe ser procesado por un hilo diferente.
- Una vez completo el proceso de indexación el índice resultante debe guardarse en un archivo en la ruta `~/indice`.
- El buscador recibirá la lista de palabras clave que se utilizarán como filtro de búsqueda.
- La búsqueda será el resultado de la conjunción de la lista de palabras clave.
- El buscador debe cargar el archivo `~/indice` en memoria para realizar la búsqueda.
- El resultado de una búsqueda será la impresión en pantalla de las rutas de los archivos que contienen la conjunción de palabras clave.

Ejemplo de uso:

```
$> ./indexador /home/usuario/tmp  
Listo! (0.3s)  
$> ./buscador hola mundo  
/home/usuario/tmp/1/1.2/1.2.3/hw.java  
/home/usuario/tmp/A/B/prueba.txt
```

Recursos

Stop words en español

<http://snowball.tartarus.org/algorithms/spanish/stop.txt>

Trie

<https://en.wikipedia.org/wiki/Trie>

Logística

El proyecto debe ser trabajado en grupos de máximo 3 personas. El proyecto debe ser entregado el día **martes 26 de mayo, 2015** a más tardar a las 11:59pm. Enviar un correo electrónico a la dirección dmm.itcr@gmail.com, el asunto debe ser los apellidos de los integrantes del grupo ordenados alfabéticamente y el sufijo "-P3", ej: RamírezSotoZamora-P3. Se debe adjuntar al correo un zip con el mismo nombre del asunto que contenga los archivos de código fuente con la solución del proyecto.

Adicionalmente se deben enviar avances semanales que contarán como un rubro más a calificar, los avances corresponden al código que se tenga al momento, no necesariamente tiene que funcionar/compilar. Los avances deben enviarse a la dirección de correo anteriormente mencionada, utilizando el mismo formato de nombre para el asunto y el archivo comprimido. Las fechas para entrega de avances son: **martes 12 de mayo, martes 19 de mayo.**