## INSTITUTO TECNOLÓGICO DE COSTA RICA II SEMESTRE 2021

**INGENIERÍA EN COMPUTACIÓN**

## IC-8060 RECUPERACIÓN DE INFORMACIÓN TEXTUAL TAREAS PROGRAMADA 2

La tarea consiste en programar una aplicación que permita indexar y buscar una colección de páginas web. Se debe usar la biblioteca [Lucene de Apache](https://lucene.apache.org/) tanto para crear el índice de la colección como para realizar las búsquedas. Se quiere aprovechar además la capacidad de Lucene para indexar partes de los documentos en “campos” independientes que pueden ser usados explícitamente en las búsquedas. Hay disponible un [tutorial](http://www.lucenetutorial.com/) muy rápido para Lucene.

La colección completa consiste de un solo archivo que contiene un subconjunto de las páginas HTML de Wikipedia en español. La idea es probar la aplicación con versiones de diferentes tamaños de la colección completa.

Cada versión consiste de un único archivo que contiene muchas páginas HTML. Cada página web empieza con la línea siguiente:

**<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">**

Luego de esa línea viene una línea que empieza con los caracteres "**<html**". La página termina con una línea que empieza con los caracteres "**</html>**". En esa misma línea, puede haber caracteres después de la etiqueta **</html>**, pero esos caracteres no son parte de la página web que terminó ni de la siguiente página web.

Se hará un procesamiento limitado de la estructura. Interesa indexar el contenido textual (sin etiquetas HTML) del elemento **<body>**. Además interesa indexar por aparte el texto incluido en los siguientes elementos:

|  |  |
| --- | --- |
| **Elemento** | **Texto de interés** |
| **<a>** | Texto que aparece dentro de ese elemento:  <a href=...> ***texto de interés***</a> |
| **<h1>**  **...**  **<h9>** | Texto que aparece dentro de ese elemento:  <h1> ***texto de interés***</h1>  <h5> ***texto de interés***</h5> |
| **<title>** | Texto que aparece dentro de ese elemento que está dentro de <head>:  <title>***texto de interés***</title> |

Se considerarán como términos de búsqueda las secuencias consecutivas de letras del alfabeto español o dígitos. Las secuencias deben contener al menos una letra. El caracter '\_' se aceptará como letra válida.Los números solos, las letras de otros alfabetos y otros caracteres no serán tomados en cuenta en la indización.

Ejemplos  
 válidos: alfabeto pedigüeñería nivel\_alto Algol60 B13a  
 inválidos: 2021 **ほ ホ**.

**Indexado**

La colección debe ser procesada extrayendo la siguiente información:

|  |  |
| --- | --- |
| **Elemento** | **Procesamiento** |
| **<body>** | * Todo el texto contenido en este elemento será indexado en un campo llamado ***texto***. * El campo ***texto*** será el campo default en las consultas. * Separar el texto de este elemento en palabras (secuencias consecutivas de los caracteres [A-Za-zÁÉÍÓÚÜáéíóúüÑñ]. * Eliminar stopwords (más abajo se especifica lista de stopwords). * Aplicar stemming a las palabras (se supone que el algoritmo de stemming convertirá las letras a minúsculas y eliminará las tildes) |
| **<a>** | * El contenido de este elemento será indexado en un campo llamado ***ref***. Para este campo, no se aplicará stemming a las palabras; solo se convertirán a minúsculas y se eliminarán las tildes. Se debe preservar la eñe en su forma original. |
| **<h?>** | * El contenido de este elemento será indexado en un campo llamado ***encab***. Se le aplicarán las mismas operaciones que para el elemento <body>. |
| **<title>** | * El contenido de este elemento será indexado en un campo llamado ***titulo***. Se le aplicarán las mismas operaciones que para el elemento <a>. |
| **href="..."** | * El contenido de este atributo del elemento **<a>** **NO** será indexado, pero será almacenado en un campo llamado ***enlace***. * Solo interesan los enlaces que empiezan con el patrón "**../../../../articles/**". * Solo debe almacenarse lo que viene luego del patrón anterior. Por ejemplo, el enlace:  "**../../../../articles/h/i/s/Historiador.html**" se almacenará como:  "**h/i/s/Historiador.html**". * Se deben eliminar los enlaces repetidos. |

Se usará la lista de stopwords que se adjunta al final de este documento. Los términos podrán ser simplificados por medio de la extracción de raíces (stemming). Se utilizará la técnica de eliminación de afijos por medio del algoritmo de Porter para español. Se recomienda usar la implementación Snowball del stemmer en español. Está disponible en Lucene así como en muchos lenguajes de programación.

La aplicación desarrollada debe poder escoger el archivo con la colección de páginas web a indexar. También se debe especificar el archivo de stopwords que se usará y si se realizará extracción de raíces o no. Luego de cada paso de indexado, deben poder realizarse búsquedas.

**Búsqueda**

Lucene provee un lenguaje de consulta muy rico, el cual puede ser usado entre otras cosas para ejecutar consultas booleanas, vectoriales, frases, por proximidad, o con errores; se puede escoger además los campos específicos del índice usado. En la página [Apache Lucene – Query Parser Sintax](https://lucene.apache.org/core/2_9_4/queryparsersyntax.html) se muestran ejemplos de los diferentes tipos de consulta que están disponibles directamente.

La aplicación debe proveer una interfaz que permita escoger un índice, leer consultas e invocar las facilidades de Lucene para realizar dichas consultas. Debe ser posible aprovechar los diferentes campos especificados.

Como resultado de las consultas, la aplicación debe mostrar los títulos (elemento <title>) de los mejores páginas del escalafón. Inicialmente se muestran los mejores 20 páginas del escalafón, pero el usuario puede pedir grupos adicionales con las siguientes 20 páginas del escalafón.

Adicionalmente, el usuario debe poder visualizar cualquier página que aparezca en el resultado que se le presenta. El sistema debe extraer la página del archivo con la colección y mostrarla en un navegador. **¡No deben almacenarse las páginas en el índice!** Se debe almacenar suficiente información en el índice para extraer **eficientemente** de la colección, el fragmento de texto que corresponde a dicha página.

Finalmente, la interfaz de consulta debe permitir mostrar, para cualquier página que aparezca en el resultado, la lista de los enlaces extraídos de dicha página (campo enlace).

**Colecciones**

Se trabajará con una colección que es una descarga de Wikipedia en español de principios de la década del 2010. La colección solo contiene el texto de las páginas web. No incluye imágenes u otros recursos multimediales.

Para eso se les proveerá con el siguiente conjunto de [subcolecciones](https://tecnube1-my.sharepoint.com/:f:/g/personal/jaraya_itcr_ac_cr/EtDIAn2k9DZHgyyRG9nkS6YBmSL__IPH7KBXDvAxQGZLkA) derivadas de la colección base, h0. Todas las páginas de una colección están contenidas en la colección siguiente. Por ejemplo, todas las páginas de h4 están contenidas en h3. Los tamaños de las colecciones crecen en un factor 2,5: |hi| = 2,5 \* |hi+1|, i=0..7.

|  |  |  |  |
| --- | --- | --- | --- |
| **Colección** | **# de páginas** | **# de líneas** | **# de bytes** |
| h0 | 1.362.474 | 201.952.031 | 20.276.602.880 |
| h1 | 544.961 | 80.950.620 | 7.875.042.797 |
| h2 | 218.263 | 32.305.477 | 3.151.840.161 |
| h3 | 87.321 | 12.895.784 | 1.261.175.583 |
| h4 | 34.872 | 5.157.141 | 505.681.477 |
| h5 | 13.880 | 2.057.194 | 203.277.020 |
| h6 | 5.526 | 826.545 | 81.099.415 |
| h7 | 2.193 | 329.527 | 32.368.999 |
| h8 | 827 | 123.710 | 12.529.664 |

En cuanto a rendimiento de indización, se considerará aceptable indizar h3 en 12 minutos o h4 en 5 minutos. Mucho mejores rendimientos son factibles, pero por la heterogeneidad del equipo disponible se usarán los valores anteriores como guía. Si se puede indizar h3 en menos de 5 minutos es factible indizar h0 en menos de una hora.

**Consultas**

La siguiente tabla muestra el resultado esperado para algunas consultas de prueba.

| **Consulta** | **Colección** | **Resultado esperado** |
| --- | --- | --- |
| magnoel | h8 | Éxito |
| titulo:magnoel | h8 | Éxito |
| encab:magnoel | h8 | Éxito |
| cartago | h8 | Fallo |
| cartago | h7 | Éxito |
| alejandro AND magno | h8 | Fallo |
| alejandro AND magno | h7 | Éxito |
| ref:alejandro AND ref:magno | h7 | Éxito |
| amintas | h7 | Fallo |
| amintas | h6 | Éxito |
| tenistas de macedonia | h6 | Fallo |
| tenistas de macedonia | h5 | Éxito |
| ref:"tenistas de macedonia" | h5 | Éxito |
| turrialba | h5 | Fallo |
| turrialba | h4 | Éxito |

Éxito: se encontró al menos un documento en la colección especificada.  
Fallo: no encontró ningún documento en la colección especificada.

**Consideraciones finales**

La tarea puede ser realizada en grupos de dos personas.

La fecha límite de entrega es el martes 26 de octubre a las 10:00pm.

**Lista de stopwords**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| a  acá  ahí  ajena  ajenas  ajeno  ajenos  al  algo  algún  alguna  algunas  alguno  algunos  allá  alli  allí  ambos  ampleamos  ante  antes  aquel  aquella  aquellas  aquello  aquellos  aqui  aquí  arriba  asi  atras  aun  aunque  bajo  bastante  bien  cabe  cada  casi  cierta  ciertas  cierto  ciertos  como  cómo  con  conmigo  conseguimos  conseguir  consigo | consigue  consiguen  consigues  contigo  contra  cual  cuales  cualquier  cualquiera  cualquieras  cuan  cuán  cuando  cuanta  cuánta  cuantas  cuántas  cuanto  cuánto  cuantos  cuántos  de  dejar  del  demás  demas  demasiada  demasiadas  demasiado  demasiados  dentro  desde  donde  dos  el  él  ella  ellas  ello  ellos  empleais  emplean  emplear  empleas  empleo  en  encima  entonces  entre  era | eramos  eran  eras  eres  es  esa  esas  ese  eso  esos  esta  estaba  estado  estais  estamos  estan  estar  estas  este  esto  estos  estoy  etc  fin  fue  fueron  fui  fuimos  gueno  ha  hace  haceis  hacemos  hacen  hacer  haces  hacia  hago  hasta  incluso  intenta  intentais  intentamos  intentan  intentar  intentas  intento  ir  jamás  junto | juntos  la  largo  las  lo  los  mas  más  me  menos  mi  mía  mia  mias  mientras  mio  mío  mios  mis  misma  mismas  mismo  mismos  modo  mucha  muchas  muchísima  muchísimas  muchísimo  muchísimos  mucho  muchos  muy  nada  ni  ningun  ninguna  ningunas  ninguno  ningunos  no  nos  nosotras  nosotros  nuestra  nuestras  nuestro  nuestros  nunca  os | otra  otras  otro  otros  para  parecer  pero  poca  pocas  poco  pocos  podeis  podemos  poder  podria  podriais  podriamos  podrian  podrias  por  por qué  porque  primero  puede  pueden  puedo  pues  que  qué  querer  quien  quién  quienes  quienesquiera  quienquiera  quiza  quizas  sabe  sabeis  sabemos  saben  saber  sabes  se  segun  ser  si  sí  siempre  siendo | sin  sín  sino  so  sobre  sois  solamente  solo  somos  soy  sr  sra  sres  sta  su  sus  suya  suyas  suyo  suyos  tal  tales  también  tambien  tampoco  tan  tanta  tantas  tanto  tantos  te  teneis  tenemos  tener  tengo  ti  tiempo  tiene  tienen  toda  todas  todo  todos  tomar  trabaja  trabajais  trabajamos  trabajan  trabajar  trabajas | trabajo  tras  tú  tu  tus  tuya  tuyo  tuyos  ultimo  un  una  unas  uno  unos  usa  usais  usamos  usan  usar  usas  uso  usted  ustedes  va  vais  valor  vamos  van  varias  varios  vaya  verdad  verdadera  vosotras  vosotros  voy  vuestra  vuestras  vuestro  vuestros  y  ya  yo |