A. Contexto

El proyecto Gutenberg (http://www.gutenberg.org) contiene más de 20.000 libros que pueden ser accedidos de manera gratuita a través de la red. Buscar por contenido en esta enorme base es una característica muy atractiva, aunque por ahora sólo está disponible de forma experimental. Por ejemplo, la búsqueda de "En un lugar de La Mancha" puede tardar por encima de los 20 segundos (aunque la segunda vez que se repita irá mucho más rápido).

El objetivo de esta práctica es crear un buscador bibliográfico por contenido, que permita encontrar de manera eficiente los párrafos y capítulos de libros que incluyen ciertas palabras, ya sea algunas de ellas (búsqueda con OR), todas ellas (búsqueda con AND), o las palabras seguidas en el mismo orden.

B. El Problema

Analizar, diseñar e implementar un buscador bibliográfico por contenido. El programa admitirá una serie de comandos, que se leerán siempre de la entrada estándar, produciendo el resultado en la salida estándar. Los comandos admisibles son los siguientes:

- **Insertar un nuevo libro**: indicando el ISBN, título, autor, año, número de capítulos y el contenido en sí del libro. Los libros se dividen en capítulos, y estos a su vez en párrafos. El formato de entrada se describe más adelante.
- **Buscar palabras con AND**: listar todos los párrafos de libros donde aparecen todas las palabras dadas. Las palabras se definen como cualquier sucesión de una o más letras delimitadas por caracteres que no sean letras.
- **Buscar palabras con OR**: listar todos los párrafos de libros donde aparecen algunas de las palabras dadas.

El programa diseñado deberá cumplir los siguientes requisitos:

- Las búsquedas deben ser independientes de mayúsculas/minúsculas y de las tildes.
- Se requiere que no se pierda ni una sola búsqueda para aprobar la práctica.
- El programa debe estar bien diseñado para conseguir la máxima eficiencia de tiempo y de memoria (con especial hincapié en lo primero). Por ejemplo, no será admisible almacenar y recorrer todos los libros para cada búsqueda nueva.

C. Comandos adicionales

Estos comandos adicionales serán voluntarios para los alumnos que se acojan al sistema de evaluación continua. Para los restantes, serán obligatorios.

- **Eliminar un libro dado**: se deben suprimir todas las referencias al libro y a las palabras que contiene, liberando la memoria correspondiente.
- **Buscar palabras CONSECUTIVAS**: mostrar todos los párrafos de libros donde aparecen todas las palabras dadas y de forma consecutiva.
- **Buscar en libros y en capítulos**: similar a las búsquedas con AND y con OR, pero en capítulos o libros completos; es decir, buscar los capítulos (o los libros) que tenga todas, o algunas, de las palabras dadas. Se deberá evitar la duplicación de información, en relación a la búsqueda de párrafos.
- **Buscar todos los libros de un autor**: dado el nombre del autor, listar los libros del mismo. Se debe usar una estructura que permita una consulta eficiente.

D. Formato de entrada

La entrada está compuesta de varios comandos. Todos los comandos están en una línea distinta, excepto la inserción, que ocupará varias. Los comandos admisibles son: **i** (para insertar), **a** (búsqueda con AND), **o** (búsqueda con OR), **s** (salir). La entrada acabará siempre con un comando **s**. El formato de los comandos es el siguiente:

- **Insertar**: después de la **i** aparecerán tres enteros, *B*, *A* y *C* (separados por espacios) que indican el ISBN del libro (suponer un entero de 9 dígitos), el año de publicación y el número de capítulos del libro. La siguiente línea contiene el título del libro, y la siguiente el nombre del autor. A continuación vienen los capítulos. Cada capítulo puede ocupar una o varias líneas, y acaba con la palabra clave "FinDeCapitulo", que no aparecerá en el texto (y es independiente de mayúsculas, minúsculas o tildes). En cada capítulo puede haber uno o más párrafos. Los párrafos están separados por una línea en blanco. Tanto los párrafos como los capítulos se numeran automáticamente a partir de 1. Este puede ser, por ejemplo, un comando **i** válido:

```
i 283930182 2007 2
El Quijote (versión mini)
GinesGM & Miguel de Cervantes
Capítulo 1. QUE TRATA DE LA CONDICIÓN Y EJERCICIO DEL QUIJOTE
En un lugar de la Mancha, de cuyo nombre no quiero acordarme...
(ojo, algunos listillos dicen "no puedo acordarme"), vivía el Quijote. FinDeCapitulo
Capítulo 2. QUE TRATA DE LO QUE VERÁ EL QUE LO LEYERE
- ¡Aquí fue Troya! ¡Aquí mi desdicha, y no mi cobardía, se llevó mi gloria!

- Vayan a estudiar a Salamanca. Todo es burla sino estudiar y más estudiar…
FINDECAPITULO
```

- **Búsqueda con AND**: después de la letra **a**, aparecerá una lista de una o más palabras que se buscan. Por ejemplo:

```
a el LISTILLOS del quijote él
a llevo la Condición
```

- **Búsqueda con OR**: después de la letra **o**, aparecerá una lista de una o más palabras que se buscan. Por ejemplo:

```
o condicion
o TROYANO TROYA
```

E. Formato de salida

Después de cada comando, se mostrará por pantalla información sobre el resultado del mismo. La salida tendrá el siguiente formato:

- **Insertar**: la salida serán 5 líneas, que contendrán: (1) el título del libro insertado, (2) el autor y año (separados por coma), (3) el número total de capítulos leídos, (4) el número total de párrafos, y (5) el número total de palabras, ya sean repetidas o no. Por ejemplo, la salida para el ejemplo de arriba será:

```
El Quijote (versión mini)
GinesGM & Miguel de Cervantes, 2007
2 capítulos
3 párrafos
70 palabras
```

- **Búsqueda con AND y con OR (también las adicionales)**: los resultados irán numerados de forma consecutiva, y para cada uno se indicará: el título, el autor(es), el año, el número de capítulo, y el número de párrafo, separados por comas. La lista estará ordenada por ISBN, y en caso de empate por número de capítulo y número de párrafo (siempre de menor a mayor). La última línea contendrá el número total de resultados. Por ejemplo, el resultado de la búsqueda "o capitulo" en el ejemplo de arriba sería:

```
1. El Quijote (versión mini), GinesGM & Miguel de Cervantes, 2007, Cap. 1, par. 1
2. El Quijote (versión mini), GinesGM & Miguel de Cervantes, 2007, Cap. 2, par. 1
Total: 2 resultados
```

Se deja libertad a los alumnos para decidir el formato de los comandos adicionales no especificados (eliminar libro y buscar los libros de un autor).

F. Fases de desarrollo

Para una correcta resolución de la práctica, los alumnos deberán cumplir las siguientes fases en los plazos señalados abajo. El cumplimiento de estas fases se validará en el juez on-line de la asignatura, superando los problemas indicados entre paréntesis. Además, se entregará una breve memoria del trabajo realizado en cada fase. Las fases de desarrollo son:

- **F1.** Hasta el 12 de noviembre: resolver los problemas básicos referentes a normalizar un texto (002 y 003), separar las palabras (004), definir el tipo **Libro** (005), el tipo **Aparición** (006), y el tipo **ListaEntero** de listas ordenadas de enteros (007).
- **F2.** Hasta el 5 de diciembre: implementar el tipo **Lista<T>** de listas genéricas ordenadas, instanciado a listas de apariciones (008), y un tipo **HashLibros** de diccionarios representados con tablas de dispersión (abierta o cerrada) de enteros en libros (204). Opcionalmente, y como pasos previos al ejercicio 204, se pueden implementar tablas de dispersión de cadenas en naturales (202) y luego de forma genérica, **Hash<C, V>** (203).
- **F3.** Hasta el 21 de diciembre: crear un tipo diccionario implementado con árboles (ya sea trie, AVL o B) donde la clave son cadenas y el valor es genérico, **Arbol<V>** (301). Instanciar a Arbol<Lista<Aparición> >, y crear la operación **cargarLibro**, que mete las apariciones de palabras de un libro en el árbol (302).
- **F4.** Hasta el 18 de enero: definir las operaciones **buscarAND** (303) y **buscarOR** (304), y completar la práctica (305). Entrega final de la memoria de la práctica.

En todos los casos, se podrán realizar los envíos al juez on-line hasta las 14:30:00 del día señalado, siendo la entrega de la memoria a lo largo de todo el día.

Los grupos que cumplan todos los plazos señalados y de forma satisfactoria, tendrán un +1 en la nota final de la práctica. Los grupos que incumplan uno de los plazos, tendrán que hacer uno de los comandos adicionales descritos en el apartado C (a elegir); los que incumplan dos plazos harán dos adicionales; y los que incumplan tres plazos harán tres adicionales. Los alumnos que entreguen la práctica en junio o en la convocatoria de septiembre, deberán hacer todos los comandos adicionales.

G. Documentación

La documentación a entregar durante las fases intermedias de desarrollo (**F1**, **F2** y **F3**) contendrá: una descripción somera de los aspectos más relevantes del trabajo realizado, el listado del código, y en su caso, una indicación de los envíos realizados al juez on-line.

La documentación final (fase **F4**) contendrá los siguientes apartados:

1. **Portada**. Nombre de los alumnos y e-mail de cada uno.
2. **Análisis del problema**. Encontrar los tipos abstractos que aparecen en el problema, y en qué partes aparecen. Analizar las diferentes alternativas que se presentan para la implementación de esos tipos.
3. **Diseño de la aplicación**. Mostrar un esquema gráfico global de la estructura de tipos de datos existentes. Detallar la descomposición modular del programa, qué módulos existen, cuál es la responsabilidad de cada uno y la relación de uso. Documentar cualquier otra decisión de diseño que pueda resultar de interés.
4. **Listado del código**. Incluyendo el fichero `makefile` necesario para compilar.
5. **Informe de desarrollo**. Describir cómo ha sido la coordinación y el reparto del trabajo entre los miembros del grupo. Rellenar las tablas de dedicación personal en las distintas fases del trabajo. Se utilizarán tablas como las explicadas en las páginas 37 y 350 del texto guía, rellenadas con el mayor rigor posible.
6. **Conclusiones y valoraciones personales**.

H. Evaluación de la práctica

H.1. Obligatorio

Para aprobar la práctica se requiere que:

- El programa se pueda ***compilar sin errores*** en las máquinas del laboratorio de prácticas, en la fecha y hora en la que se realice la entrevista final con los alumnos. En particular, el programa estará escrito en C++, y el código se deberá compilar en Linux.
- El programa debe ***funcionar correctamente***, sin colgarse y produciendo ***resultados correctos*** para el conjunto de pruebas que se determinen. Para ello, el profesor puede usar (pero no está limitado a) los casos de prueba incluidos en el juez on-line de la asignatura.
- La ***memoria de la práctica*** debe contener todos los puntos indicados en el apartado G, y debe ser entregada en el plazo que se establezca. ¡La documentación entregada no debe contener *faltas de ortografía* (incluida la omisión de tildes)!

H.2. Criterios de valoración

La práctica se puntuará de acuerdo con los siguientes criterios de calidad del software:

- **Análisis y diseño**. Se valorará la calidad y adecuación del diseño y el análisis realizados, y la dedicación a estas fases previas a la implementación. Se deben encontrar los tipos abstractos que aparecen, e implementarlos usando clases, eligiendo las estructuras más adecuadas.
- **Modularidad.** La funcionalidad debe estar bien repartida entre los módulos. Debe estar claro el sentido y la responsabilidad de cada módulo. Se debe respetar el principio de ocultación de la implementación.
- **Uso del lenguaje**. El código deber ser claro, legible, robusto y eficiente. No crear procedimientos muy largos y complejos. Se valorará el uso de clases genéricas (plantillas) y precondiciones / postcondiciones (asertos).
- **Seguimiento continuo**. El correcto cumplimiento de las fases de desarrollo, marcadas en el apartado F, será un aspecto a favor en la evaluación de la práctica. Se desaconseja la posibilidad de no seguir los plazos a cambio de realizar comandos adicionales.

H.3. Otras cuestiones

La práctica se deberá realizar preferiblemente en **grupos de dos alumnos**. De forma extraordinaria se permiten **grupos de 1 alumno**, pero no se prevé ninguna reducción del trabajo para los mismos.

Para realizar pruebas y para la verificación de las fases de desarrollo, los profesores dejarán en la página web del juez on-line (http://dis2.um.es/~mooshak), dentro del concurso "AED: Practicas", los problemas mencionados en el apartado F. Cada alumno dispondrá de un *login* y *password* para acceder a este sistema; el grupo deberá elegir y utilizar una de las cuentas para hacer los envíos al juez.

La fecha de entrega definitiva de esta práctica coincide con la entrega de la última fase, es decir, el 18 de enero de 2008. La forma de hacer las entregas (en papel, por email, a través de SUMA, etc.) lo indicará cada profesor de prácticas.

AVISO IMPORTANTE

Las prácticas de todos los grupos, en todas las convocatorias y titulaciones, serán sometidas a un sistema computerizado de **detección de plagios** (ver http://aps.arxiv.org/PS_cache/cs/pdf/0703/0703134v4.pdf). Copiar la práctica de otro grupo supondrá el suspenso fulminante de la asignatura en la convocatoria correspondiente, para todos los grupos implicados.