Informe de Prácticas de Fundamentos de Bases de Datos

Práctica 3

Rodríguez Marrero, David

Larriba Moreno, Sergio

Índice

* Introducción 2
* Operaciones que implementar
  + Función createTable (replaceExtensionByIdx) 3
  + Función createIndex 4
  + Función printTree 5
  + Función findKey 6
  + Función addIndexEntry 7
  + Función addTableEntry 8
* Interfaz de Usuario 9
* Conclusiones 10

Introducción

La práctica se va a centrar en cómo el ordenador lee y entiende las tablas que almacenamos en una base de datos. Esta práctica se va a centrar en tablas, y cómo se gestionan con un fichero de datos en el que se guarda la información y otro de índices que ayuda a buscar una información en concreto según un árbol binario implementado.

Por otra parte, también vamos a hacer uso de las funciones fwrite, fread y fseek que nos ayudaran a gestionar las búsquedas e inserciones en los archivos de datos e índices.

Han incluido en la práctica un zip con scripts para resolver y comprobar el resultado del menú que hemos implementado. Gracias a esos scripts podremos ver qué sentencias tenemos que imprimir en consola para un buen entendimiento del programa.

Al final de la práctica, entregaremos todos los códigos de las consultas y el Makefile que hemos utilizado prácticamente igual al dado en Moodle. También entregaremos este informe en el cual hemos redactado nuestros pensamientos durante toda la práctica.

CreateTable

A screenshot of a computer

Description automatically generated with medium confidence

A screenshot of a computer

Description automatically generated with medium confidence

Esta función abre el fichero con nombre tableName que se pasa por argumento y si no lo puede abrir lo crea e inicializa la cabecera con un -1 que indica que el listado de registros borrados está vacío.

Después de esto, creamos el archivo de índices con mismo nombre que el archivo anterior pero con formato .idx que cambiamos con la función implementada replaceExtensionByIdx.

Luego, llamamos a createIndex, liberamos memoria y retornamos si ha salido todo bien.

CreateIndex

A screenshot of a computer

Description automatically generated with medium confidence

Esta función empieza creando el fichero de índices si no existe y escribe dos -1 en la cabecera los cuales indican la raíz del árbol y el nodo borrado.

PrintTree

A screenshot of a computer

Description automatically generated with medium confidence

Nuestra función printTree mira si el fichero está vacío, es decir que la raíz sea -1 y si no, llama a la función recursiva printnode la cual lee los nodos del archivo de índices y los va imprimiendo recursivamente teniendo en cuenta su profundidad hasta encontrar un -1 en node\_id que significará que no existe ese nodo.

Text

Description automatically generatedText

Description automatically generatedFindKey

En esta función buscamos un book\_id y devolveremos el offset de la posición en la que se encuentra mediante argumentos. Primero abrimos el archivo de índices e inicializo ese offset de salida al de la raíz. Después busco el nodo que ocupa la posición pos que va a ir cambiando si no lo encontramos y lo guardo en node y mediante comparación vamos recorriendo todo el árbol binario según si los bytes son mayores o menores. Si llegamos a un valor -1, es que el nodo buscado no existe, si lo encontramos, guardamos el offset en la variable del argumento y retornamos.

A screenshot of a computer

Description automatically generatedAddIndexEntry

En esta función queremos introducir una clave por lo que primero miramos a ver si ya existe, si no, abrimos el archivo de índices y miro si hay registros borrados, si no hay añado el nuevo nodo, si hay, leo el nodo borrado y actualizo el nodo de la cabecera que indica que hay nodos borrados.

Luego, inicializamos el nuevo nodo sin hijos, buscamos el padre por el offset dado e inserto el nodo respetando el árbol binario.

A screenshot of a computer

Description automatically generated with medium confidence

AddTableEntry

A screenshot of a computer

Description automatically generated

Para insertar un nuevo dato, primero miro a ver si ya existe ese dato, si no, miro si existen registros borrados, si no existen, escribimos al final del archivo la información y después llamamos a la función anterior para editar el fichero de índices.

Interfaz de Usuario

Hemos creado un menú casi idéntico estructuralmente al de la práctica 2 en el que el usuario tiene que elegir use, insert o print para hacer distintas cosas en torno a la creación de tablas inserción en la tabla o impresión del árbol de índices.

* Use: El sistema preguntará por el nombre de la tabla a utilizar con el formato correcto y llamara a la función createTable.
* Insert: El sistema preguntará por la clave y el título para almacenar y llamará a la función addTableEntry para introducirla en los dos ficheros.
* Print: Se muestra el árbol binario de búsqueda por pantalla. Se debe mostrar un mensaje de error si no se ha seleccionado una tabla antes usando use.

Para usar los comandos Insert y Print se debe pasar primero por Use ya que si no no habrá ninguna tabla creada.

Nuestro menú también dispone de una cuarta entrada posible que es el exit la cual sale del programa.

Gracias a este programa implementado, podemos hacer uso de nuestras funciones más fácilmente que si tuviésemos que hacer un main como el tester pero llamando a todas las funciones con unos valores pedidos por pantalla. Ayuda bastante repartirlo en un menú en el que puedes ver la información que quieres, ni más ni menos.

Conclusiones

En esta práctica hemos aprendido cómo el ordenador lee las tablas que llevamos usando durante las dos últimas prácticas y como a través de bytes es posible trasmitir, guardar o editar una información concreta.

Por otra parte, también hemos entendido la dificultad que tienen los programas que hacen estas comunicaciones con el ordenador directamente con mucha más información de la que nosotros tenemos que gestionar en nuestra práctica.

Hemos adquirido nuevos conocimientos del lenguaje C usando funciones como el fwrite, fread y fseek que son fundamentales en el uso de archivos. También hemos entendido las diferencias entre un archivo de información y otro de índices el cual gestiona un árbol binario por lo que estamos tratando con estructuras de datos vistas y aprendidas anteriormente que nos han ayudado a comprender mejor la práctica.

También hemos aprovechado lo aprendido en la práctica 2 ya que nos ha facilitado el menú y hemos entendido que con un programa es mucho más fácil comunicarse con las funciones ya que a lo largo de la práctica nos hemos perdido a la hora de ejecutar los test individualmente para ver si nuestras funciones funcionaban o no.

Finalmente, todas estas prácticas nos han introducido por completo al mundo de la gestión de bases de datos desde lo más externo como es el lenguaje SQL hasta lo más interno como es comprender qué lee el ordenador y como lo gestiona.