Actividades\_1.2

1. **Investigue y explique con sus palabras en qué consiste el método de búsqueda binaria en ficheros**

La búsqueda binaria es un algoritmo eficiente para encontrar un elemento en una lista de elementos. Consiste en dividir continuamente a la mitad la cantidad de la lista que puede contener al elemento y reducir hasta obtener una. Este algoritmo reduce el tiempo de búsqueda considerablemente.

1. **Realice un ejemplo de búsqueda secuencial y binaria en clase suponiendo que tiene que acceder a un valor dentro de un conjunto ordenado de valores. Compute y compare el número de lecturas en ambos procesos para varios valores de búsqueda**

En la organización secuencial los registros se colocan secuencialmente uno detrás de otro, cada registro nuevo se añadiría al final de la lista, por lo que la búsqueda es complicada ya que habría que ir bloque a bloque hasta encontrar el que queremos. Un ejemplo sencillo para la búsqueda secuencial es siempre que tengamos una lista de registro no muy extensa podríamos ir mirando uno a uno hasta encontrar el que queremos.

Ejemplo: Queremos encontrar un habitante concreto en una población de 200 habitantes.

Esto es eficiente ya que la población es poca pero en una población con miles y miles de habitantes nos resultaría más sencillo creando un fichero índice el cual contiene información de la posición que ocupa cada registro.

Un ejemplo de búsqueda binaria es si tengo un conjunto de 200 valores y quiero encontrar el 130, buscas en el 100 ya que es la mitad, y así habría que ir reduciendo a la mitad hasta encontrar el deseado, cada vez que recortemos a la mitad será una nueva lectura. Para ello habría que ordenarlo por el campo de búsqueda. Para el siguiente ejemplo: Vamos a buscar “Leandro”

|  |  |  |  |
| --- | --- | --- | --- |
| ID | NOMBRE | LOCALIDAD | EDAD |
| 1 | Luis | Villamartín | 19 |
| 2 | Alfonso | Ubrique | 25 |
| 3 | Alberto | Jédula | 14 |
| 4 | Leandro | Prado del Rey | 35 |
| 5 | Carlos | Ubrique | 20 |

Si queremos buscar a Leandro mediante una búsqueda secuencial tendríamos que realizar 4 lecturas ya que habría que ir leyendo y comparando desde el principio, mientras que en la binaria tendríamos que hacer 2 lecturas ya que leeríamos en la mitad que es 3, al no ser volvemos a coger el de la mitad desde el 3 hasta el último que sería el 5, la mitad en este caso es 4,por lo que encontraríamos el que estamos buscando.

**3. ¿Cuántos índices primarios y de agrupamiento puede tener un fichero ordenado?**

Un fichero ordenado solo puede tener un índice primario ya que cada fichero solo tiene un campo clave que es el de I y todos los índices de agrupamiento que queramos siempre que tenga sentido, es decir cuando según el criterio haya valores que se repitan.

**4.Comente ventajas e inconvenientes respecto a la actualización de datos en ficheros con organización tipo hash.**

Una de las principales ventajas es que ofrece mayor seguridad, ya que los valores hash no pueden convertirse en valores originales sin una clave, es importante para la gestión y seguridad de base de datos. Además el acceso a la base de datos es más rápido. Los valores hash robados no son útiles para los atacantes sin las aplicaciones apropiadas.

Uno de los inconvenientes es la necesidad de ampliar el espacio de la tabla si el volumen de datos crece, es una operación costosa y es difícil para recorrer todos los elementos.

**5.Si tenemos un archivo de datos de 2.000 jugadores con tamaño fijo de 80 bytes y un disco de tamaño de bloque igual a 1.024 bytes, determine el número de bloques requerido y el coste de una búsqueda binaria en cuanto a número necesario de accesos a bloques para encontrar un registro de datos.**

Registro lógico = 80 bytes

Registro físico = 1024 bytes

Factor bloque = 1024/80 = 12.8 = 12 registros por bloque

Para calcular el número de bloques requeridos dividiremos el total de jugadores entre los jugadores que entran por bloque.

Bloques requeridos = 2000/12 = 166,6 bloques, por lo que necesitaríamos 167 bloques.

Para la búsqueda binaria hay dividir el conjunto de registros a la mitad hasta encontrar el registro deseado.

Costo de la búsqueda binaria = log2 (167) = 7.38 = 8 lecturas habría que realizar

**6.Suponga que en el ejercicio anterior creamos un índice formado por la clave primaria (5 bytes) y un apuntador de 4 bytes. ¿Cuántas entradas de índice tendremos? ¿Cuántos accesos a bloques de disco necesitaremos ahora para efectuar una búsqueda binaria?**

Registro lógico = 9 bytes ya que la clave primaria son 5 bytes y el apuntador 4 bytes.

Registro físico = 1024

Factor bloque = 1024/9 = 113,7 = 113 registros por bloque

Número de bloques = 2000/113 = 17.7 = 18 bloques necesitaremos

Para efectuar la búsqueda binaria: log2 (18) = 4.17 = 5 lecturas

Tendríamos 2000 entradas de índice, una por cada registro

**7.¿Qué problemas observa al usar ficheros de índices primarios en ficheros ordenados, respecto a la inserción y eliminación de registros?**

Uno de los principales problemas que encontramos es que cuando vamos insertando o eliminando registros del archivo hay que actualizar el índice, lo que nos llevara más tiempo. Además la inserción y eliminación de registros pueden resultar en un espacio no utilizado del archivo principal. Otro de los problemas que encontramos en la inserción es que puede provocar una fragmentación en el archivo principal, es decir que pueden no estar almacenados de forma contigua.

**8.Investigue la diferencia entre una estructura de índice tipo árbol ByB+.**

La principal diferencia entre ambos tipos de árbol es que, en el árbol B, los datos se pueden almacenar tanto en el nodo interno como en el de la hoja, mientras que en el B+ solo se pueden almacenar en el nodo de hojas. En el árbol B cualquier nodo puede tener más de dos hijos mientras que el B+ es un árbol con un nodo que consiste en un gran número de hijos por nodo.

Un árbol B+ es lo mismo que un árbol B, pero en el árbol B+ hay un nivel adicional agregado.