

[dbd](#)

[archivo](#)

[acceso a los archivos](#)

[Secuencial físico:](#)

[Secuencial indizado \(lógico\):](#)

[Directo:](#)

[tipos de archivos](#)

[serie:\(archivos\)](#)

[secuencial:\(árboles\)](#)

[directo: \(hash\)](#)

[¿Cómo se organiza un archivo?](#)

[archivos estructurados \(archivos binarios\)](#)

[campos de longitud variable](#)

[registros](#)

[NRR](#)

[Fragmentación](#)

[Eliminación](#)

DBD

- Es una colección de datos relacionados.
- Colección de archivos diseñados para servir a múltiples aplicaciones
- Un dato representa hechos conocidos que pueden registrarse y que tienen un resultado implícito.

Para que estos datos persistan, se deben guardar en una memoria secundaria.

Propiedades implícitas de una Base de Datos

- Una BD representa algunos aspectos del mundo real
- Es una colección coherente de datos significados inherentes.

Un conjunto aleatorio de datos no puede considerarse una BD. O sea, los datos deben tener cierta lógica.

- Se diseña, construye y completa de datos para un propósito específico. Está destinada a un grupo de usuarios concretos y tiene algunas aplicaciones preconcebidas en las cuales están interesados los usuarios.
- Está sustentada físicamente en archivos en dispositivos de almacenamiento persistente de datos.

Archivo

- Colección de registros guardados en almacenamiento secundario
- Colección de datos almacenados en dispositivos secundarios de memoria
- Colección de registros que abarcan entidades con un aspecto común y originadas para algún propósito particular

Memoria RAM: El tiempo de acceso es de micro o nano segundos.

Memoria secundaria: El tiempo de acceso es de milisegundos.

Acceso a los archivos

- **Secuencial físico:**
 - ❖ acceso a los registros uno tras otro y en el orden físico en el que están guardados
- **Secuencial indizado (lógico):**
 - ❖ acceso a los registros de acuerdo al orden establecido por otra estructura
 - ❖ • Ej: una guía telefónica, o índice temático del un libro
- **Directo:**
 - ❖ se accede a un registro determinado sin necesidad de haber accedido a los predecesores

Tipos de archivos

★ serie:(archivos)

- cada registro es accesible solo luego de procesar su antecesor, simples de acceder
- Sin orden

★ secuencial:(árboles)

- son accesibles en orden de alguna clave
- Con orden

★ directo: (hash)

- se accede al registro deseado

Archivos físicos: Como aparecen en el disco rígido. A cargo del sistema operativo.

Archivos lógicos: Como se manipulan desde el lenguaje de programación.

Operaciones

★ Operaciones básicas

- Leer(read) -> Read(nom_lógico,variable)
- Escribir(write) -> Write(nom_lógico,variable)
- Definir(assign) -> Assign(nom_lógico,nom_físico)
- Crear un archivo -> Rewrite
- Ver o modificar archivo existente -> Reset
- Cerrar archivo -> Close(nom_lógico)

Las operaciones read y write, leen o escriben sobre los buffers relacionados a los archivos. No se realizan directamente sobre el disco rígido.

En ambos casos, la variable debe ser del mismo tipo que los elementos que se declaran como parte del archivo.

★ Operaciones adicionales

- EOF(nom_lógico);(Función) -> Retorna verdadero o falso, cuando el puntero este señalando el último archivo.
 - ❖ Fin de archivo
 - ❖ ¿Cómo trabaja? -> Hay que preguntar primero,
- FileSize(nom_lógico);(Función) -> Retorna valor numérico, cantidad de registros que tiene el archivo
 - ❖ Tamaño del archivo
- Seek(nom_lógico, pos);(Procedimiento) -> Genera un acceso directo de un archivo (entre 0 y n-1)
 - ❖ Ir a una posición del archivo
 - ❖ La posición se cuenta siempre desde el comienzo del archivo
 - ❖ EL primer lugar es el 0

Siempre que se hace la operación read o write, el puntero avanza

¿Cómo se organiza un archivo?

➤ Secuencia de bytes

- Archivos de texto
- No se puede determinar fácilmente comienzo y fin de cada dato.
- Se leen o recuperan caracteres.
- No hay formato previo

- Una palabra se termina por un conjunto de caracteres que termina con blanco. Pero eso es una “convención”.
- Tipos primitivos
- Conjunto de registros
 - Hay estructura.
 - Los registros pueden estar conformados por campos.
- Organización:
 - **Registro:** Conjunto de campos agrupados que definen un elemento del archivo.
 - **Campo:** Unidad más pequeña lógicamente significativa de un archivo.

Archivos estructurados (archivos binarios)

- **Registros**
- **Longitud fija o variable**
- **Campos**
- **Longitud fija o variable**

Campos de longitud variable

- Identidad de campos: variantes, pro y contras.
- **Longitud predecible** (long. Fija), desperdicio de espacio, si el tamaño es pequeño al agrandarlo se podría desperdiciar más espacio)
- **Indicador de longitud** (al ppio de cada campo)
- **Delimitador al final de cada campo** (carácter especial no usado como dato)

Registros

- Organización de registros
- **Longitud predecible** (en cant. de bytes o cant. de campos)
- Campos fijos o variables
- **Longitud variable**
- **Indicador de longitud** (al comienzo, indica la cant. de bytes que contiene)
- **Segundo archivo** (mantiene la info de la dirección del byte de inicio de cada registro)
- **Delimitador** (carácter especial no usado como dato)
- **Long. Predecible de registros**

NRR

- Indica la posición relativa con respecto al principio del archivo
- Solo aplicable con registros de longitud fija) <- ? DICE ESO Y EN EL AUDIO DICE QUE SI SE PUEDE CON REG DE LONG VARIABLE

Fragmentación (hacer que el archivo ocupe más lugar del que necesita)

fragmentación interna: lugar vacío en el mismo registro

fragmentación externa: lugares vacíos entre registros

Eliminación (en registros de long variable):

Primer ajuste (más rápido, eventualmente genera fragmentación interna): se selecciona la primer entrada de la lista de disponibles, que pueda almacenar al registro, y se le asigna al mismo.

•**Mejor ajuste(genera fragmentación interna):** elige la entrada que más se aproxime al tamaño del registro y se le asigna completa.

•**Peor ajuste(genera fragmentación externa):** selecciona la entrada más grande para el registro, y se le asigna solo el espacio necesario, el resto queda libre para otro registro

Algoritmia: Básica o Clásica

- Básica: Agregar nuevos elementos.
- Clásica
 - ❖ Actualización
 - ❖ Merge
 - ❖ Corte de control

Los archivos siempre se pasan por referencia.

Maestro: Archivo que resume un determinado conjunto de datos

Detalle: Agrupa información que se utilizará para modificar el contenido del archivo maestro
En general, 1 maestro y n detalles.

Precondiciones

- Ambos archivos (maestro y detalle) están ordenados por el mismo criterio.
- En el archivo detalle solo aparecen empleados que existen en el archivo maestro.
- Cada empleado del archivo maestro, a lo sumo, puede aparecer una vez en el archivo detalle.
- Para hacer un read, primero hay que preguntar si no se terminó el detalle (not eof(det))

Merge

- Involucra archivos con contenido similar, el cual debe resumirse en un único archivo.

En el merge, no existe el archivo maestro pero si el detalle.

Todos los archivos detalle tienen la misma estructura y se sigue manteniendo el orden por el mismo criterio. Lo que se hace es compactar a todos los archivos en un mismo archivo, llamado maestro.

Cuando tengo tantos archivos, lo que hay que hacer, es usar arreglos.

RAM: Es rápida y de simple acceso, pero su uso tiene algunas desventajas respecto al almacenamiento secundario.

- Capacidad limitada
- Mayor costo
- Es volátil

Almacenamiento secundario: Necesita más tiempo para tener acceso a los datos que en RAM.

- Su acceso es tan “lento” que es imprescindible enviar y recuperar datos con inteligencia.
- Al buscar un dato, se espera encontrarlo en el primer intento (o pocos)
-

1. los archivos con registros de longitud variable
 - a. ocupan menos espacio que los archivos con registros de longitud fija
 - b. ocupan mas espacio que los archivos con registros con longitud fija
 - c. ocupan el mismo espacio que los archivos con registros con longitud fija
 - d. ninguna de las anteriores
2. un algoritmo de busqueda en un archivo
 - a. es mas eficiente si el archivo esta ordenado
 - b. puede ser mas eficiente si considera como precondition que el archivo esta ordenado
 - c. es igual de eficiente si el archivo esta ordenado o desordenado
 - d. ninguna de las anteriores
3. el proceso de merge de archivos
 - a. requiere que todos los archivos esten ordenados
 - b. requiere que todos los archivos esten ordenados por el mismo criterio
 - c. puede realizarse sin los archivos ordenados
 - d. no puede realizarse sin los archivos ordenados
 - e. algunas de las anteriores
 - f. ninguna de las anteriores
4. el proceso de alta de registro por ajuste optimo
 - a. se puede realizar con registros de longitud fija
 - b. se debe realizar con registros de longitud fija
 - c. se puede realizar con registros de longitud variable
 - d. se debe realizar con registros de longitud variable
 - e. ninguna de las anteriores
5. la operacion assign
 - a. se utiliza para abrir un archivo
 - b. se utiliza para posicionarse en el primer registro del archivo
 - c. vincula el archivo logico con el archivo fisico
 - d. se utiliza para saber la longitud del archivo
 - e. algunas de las anteriores

1 → d

porque no sabemos cuánto ocupa el registro con longitud variable

2 → b

podés hacer una búsqueda secuencial, pero si sabes que está ordenado podés hacerlo más eficiente (logarítmico)

3 → c

porque podés hacer muchos seek y recorrer varias veces el archivo u ordenarlo antes de hacer el merge

4 → e

no existe "ajuste óptimo"

5 → c

6.a partir del NRR

- a.se puede acceder a un registro de un archivo fragmentado en solo un acceso
- b.se puede acceder a un registro de un archivo no fragmentado en solo un acceso
- c.se puede acceder a un registro de un archivo en solo un acceso
- d.se puede lograr acceso directo a un archivo
- e.todas las anteriores
- f.algunas de las anteriores

7.un archivo fragmentado

- a.debe compactarse para optimizar el espacio utilizado
- b.no debe compactarse para optimizar el espacio utilizado
- c.a veces puede compactarse
- d.nunca debe compactarse
- e.algunas de las anteriores

8.un algoritmo de actualizacion maestro-detalle

- a.requiere que todos los archivos tengan la misma estructura
- b.puede realizarse entre archivos con diferentes estructura
- c.requiere que los archivos esten ordenados
- d.requiere que los archivos esten desordenados
- e.algunas de las anteriores

9.un archivo serie

- a.esta ordenado
- b.puede ordenarse
- c.requiere ordenarse
- d.no requiere ordenarse
- e.no esta ordenado
- f.algunas de las anteriores

10.dado un archivo con 1000 registros

- a.siempre se puede llevar a memoria RAM para hacer busquedas eficientes
- b.no siempre se puede llevar a memoria RAM para hacer busquedas eficientes
- c.siempre puede realizarse busqueda dicotomica
- d.siempre puede realizarse busqueda dicotomica
- e.algunas de las anteriores

6 → e

fragmentado puede ser fragmentación interna en un archivo de long fija, así que se puede acceder a los registros igualmente.

7 → c

8 → b

mientras tengan un dato en común puede realizarse la actualización

9 → b

segun bertone en algún momento puede requerir ordenarse entonces la d es falsa

10 → b

no sabemos si 1000 registros caben en RAM

Un algoritmo de corte de control

- a. Permite actualizar un archivo maestro con un archivo detalle.
- b. Permite generar un archivo maestro a partir de varios archivos detalles.
- c. Permite presentar la información con un formato especial.
- d. Permite actualizar un archivo maestro con varios archivos detalles.

C

En un archivo con registros de longitud variable

- a. Puede utilizar "\$" como delimitador de fin de registro
- b. Cuando un registro se modifica utiliza siempre el mismo espacio
- c. Se puede utilizar cualquier lugar libre de un archivo para insertar un registro
- d. Siempre se utiliza la política de mejor ajuste para recuperar espacio

A

Un archivo con registro de longitud variable

- a. Puede estar ordenado por algun criterio
- b. Debe estar ordenado por algun criterio
- c. Nunca puede ordenarse por algun criterio
- d. Debe tener un carácter delimitador, por ejemplo &
- e. Algunas de las anteriores
- f. Ninguna de las anteriores

A

El proceso de baja lógica

- a. Esta diseñado para borrar un registro de un archivo
- b. No necesita que el archivo este ordenado
- c. No necesita que el archivo este desordenado
- d. Se puede aplicar a archivos con registros con longitud fija
- e. Se puede aplicar a archivos con registros con longitud variable
- f. Todas las anteriores
- g. Alguna de las anteriores

F

Cuando se utiliza un algoritmo de merge

- a. Es necesario tener más de un archivo al cual aplicarlo
- b. Es necesario que el archivo maestro sea de longitud fija
- c. Es necesario que el archivo detalle sea de longitud fija
- d. No necesariamente el archivo maestro quedará ordenado
- e. No es necesario que el archivo detalle esté ordenado
- f. Todas las anteriores
- g. Algunas de las anteriores

G (D y E). la A es media confusa xq se supone que el merge es de archivos, pero BERTONE.

Si ven preguntas repetidas, marquenlas.

Si ven otras preguntas que no están, agreguenlas.

Responder abajo de todo.

Examen 1.

1. ¿Cual de las siguientes definiciones corresponde a Archivo?:

- a. Colección de registros que abarca un conjunto de entidades con ciertos aspectos en común y organizados para un propósito particular
- b. Colección de registros semejantes almacenados en disco rígido

c. Colección de registros del mismo tipo almacenados en un dispositivo de memoria secundaria

d. Todas las opciones anteriores son correctas

2. Un archivo que maneja registros de longitud fija necesita:

a. Delimitadores que indiquen el fin de cada campo

b. Delimitadores que indiquen el fin de cada registro

c. Indicadores de longitud de registro

d. Ninguna de las opciones anteriores son aplicables

3. El proceso de baja en un archivo con registros de longitud variable

a. Puede recuperar el espacio disponible con nuevas altas

b. Puede recuperar el espacio disponible compactando periódicamente el archivo

c. Puede recuperar el espacio disponible compactando el archivo ante cada baja

d. Todas las anteriores son correctas

4. El procedimiento de alta de información en un archivo:

a. Siempre agrega información al final del archivo

b. Puede recuperar espacio dado de baja físicamente

c. Siempre recupera espacio dado de baja lógicamente

d. Ninguna de las anteriores

5. El proceso de baja lógica (sin ningún agregado de otras operaciones) de un archivo:

a. Nunca recupera espacio en disco.

- b. Siempre recupera espacio en disco.
- c. A veces recupera espacio en disco.
- d. No dispongo de información suficiente para responder la pregunta.

6. Dado un archivo

- a. Siempre necesita tener un índice asociado
- b. Un índice asociado le permite optimizar las operaciones de Alta
- c. Siempre debe estar ordenado
- d. Ninguna de las anteriores**

7. La eficiencia promedio de búsqueda en un archivo sin orden es:

- a. Orden lineal. ($n/2$)**
- b. Orden logarítmico
- c. 1. (directo)
- d. No dispongo datos para contestar la pregunta.

8. Un archivo organizado con registros de longitud variable

- a. No permite realizar bajas lógicas
- b. Optimiza la utilización de espacio en disco**
- c. No permite realizar bajas físicas
- d. Sólo acepta altas al final del archivo

10. El proceso de compactación de un archivo tiene sentido ser aplicado

- a. Luego de realizar una operación de alta
- b. Luego de realizar una operación de baja lógica**
- c. Luego de realizar una operación de baja física
- d. Luego de realizar una operación de modificación

12. Un borrado lógico en un archivo de datos

- a. Recupera inmediatamente el espacio borrado, dejando el archivo de tamaño menor.
- b. No se puede aplicar con registros de longitud variable
- c. Solo se aplica con registros de longitud fija
- d. Permite recuperar el espacio con nuevas altas**

14. La política de primer ajuste, que permite recuperar espacio borrado en un archivo

- a. Solo se aplica en archivos con registros de longitud fija
- b. Solo se aplica en archivos con registros de longitud variable
- c. Genera fragmentación interna
- d. Genera fragmentación externa
- e. Hay más de una respuesta correcta (b y c)**

15. A partir de un archivo con registro de longitud fija y de luego de algunas operaciones con el mismo:

- a. Nunca genera fragmentación
- b. Puede generar fragmentación interna**
- c. Puede generar fragmentación externa

d. Las dos últimas son correctas

16. El acceso secuencial a un archivo es

- a. **Acceso a los registros uno tras otro y en el orden físico en el que están guardados**
- b. Acceso a los registros de acuerdo al orden establecido por otra estructura
- c. Acceso a un registro determinado sin necesidad de haber accedido a los predecesores
- d. Ninguna de las anteriores

17. Un archivo que maneja registros de longitud variable necesita: //PREGUNTAR

- a. Delimitadores que indiquen el fin de cada campo
- b. Delimitadores que indiquen el fin de cada registro
- c. Indicadores de longitud de registro
- d. Todas las anteriores son válidas
- e. Ninguna de las opciones anteriores son aplicables**

18) Una colisión:

- a) siempre genera saturación
- b) requiere utilizar saturación progresiva
- c) puede producir saturación**
- d) más de una es correcta
- e) ninguna es correcta

19) La siguiente secuencia de instrucciones en un archivo de longitud fija

Assign (archivo, archivo.dat);

```
reset (archivo);  
read (archivo, dato_reg);  
dato_reg.precio:=200;  
seek (archivo, 0);  
write (archivo, dato_reg);  
close (archivo);
```

a) modifica el registro de la posición 0 del archivo

b) no realiza cambios en el archivo

c) más de una es correcta

d) ninguna es correcta

20) La instrucción ASSIGN:

a) abre un archivo en modo lectura/escritura

b) verifica si existe un archivo en memoria secundaria

c) vincula un archivo con otro en memoria secundaria

d) más de una es correcta

e) ninguna es correcta

21) La técnica de dispersión de archivos se utiliza para:

Hashing

1. Mantener el archivo físicamente ordenado.
2. Mejorar la performance de las inserciones
3. Mejorar la performance de las bajas
4. no realizar grandes desplazamientos en disco.
5. Hay más de una respuesta correcta
6. Ninguna de las opciones es correcta

22) Un índice secundario asociado a un archivo:

1. Siempre debe referenciar al índice primario
2. Siempre debe permitir acceder a los registros del archivo
3. A veces debe permitir acceder a los registros del archivo
4. Puede referenciar al índice primario
5. Ninguna de las anteriores.

24) La búsqueda binaria es aplicable a:

1. Archivos con registros de longitud variable.
2. Archivos desordenados con registro de longitud fija.
3. **Archivos ordenados con registros de longitud fija.**
4. Ninguna de las opciones es correcta.

25) Un algoritmo de actualización maestro/detalles:

1. Permite actualizar un archivo maestro con un solo archivo detalle
2. Permite presentar la información con un formato especial.
3. **Permite actualizar un archivo maestro a partir de uno o varios archivos detalle.**
4. Permite Mezclar en un único archivo los registros del archivo maestro y los registros de los archivos de detalle.
5. Ninguna es correcta.

26) La política de recuperación de espacio de **Mejor Ajuste**, en archivos con registros de longitud fija

1. A veces genera fragmentación externa
2. Siempre genera fragmentación externa
3. A veces genera fragmentación interna
4. Siempre genera fragmentación interna
5. **No corresponde utilizarla con archivos de longitud fija**

27)El proceso de corte de control

1. Actualiza el archivo maestro a partir de un archivo detalle
2. Actualiza el archivo maestro a partir de varios archivos detalle

3. Genera un único archivo, uniendo un archivo maestro con un archivo detalle
4. Genera un único archivo, uniendo un archivo maestro con todos los archivos detalle
5. **Ninguna de las anteriores**

28) Un archivo con registros de longitud variable

1. Admite una organización mediante hashing
2. Admite sólo bajas lógicas
3. Admite sólo bajas físicas
4. Admite sólo política de mejor ajuste para aprovechamiento de espacio
5. **Ninguna de las anteriores**

Pueden tener más de una correcta:

29) La política de recuperación de espacio de peor ajuste, en un archivo con registros de longitud fija:

1. Puede generar fragmentación interna
2. Genera siempre fragmentación interna
3. Puede generar fragmentación externa
4. Genera siempre fragmentación externa
5. **No corresponde**

30) Para actualizar un archivo maestro a partir de N archivos Detalles

1. **Es necesario que la información que contengan sea compatible**
2. Es necesario que todos los archivos estén ordenados
3. Es necesario solo que los archivos de Detalle estén ordenados
4. Es Necesario solo que el Maestro esté ordenado
5. Es necesario que todos los archivos tengan la misma estructura

31) Un archivo con registros de longitud fija

1. **A veces tiene fragmentación**
2. Debe tener fragmentación
3. No puede tener fragmentación
4. Ninguna de las anteriores

32) En un archivo con registros de longitud variable

1. **Puede utilizar “\$” como delimitador de fin de registro**
2. Cuando un registro se modifica utiliza el mismo espacio
3. Se puede utilizar cualquier lugar libre de un archivo para insertar un registro
4. Siempre se utiliza la política de mejor ajuste para recuperar espacio

33) Para poder realizar merge entre dos archivos:

1. Los archivos deben estar ordenados con índices implementados con alguna estructura de dato no lineal
2. Los archivos deben estar ordenados con índices implementados con alguna estructura de dato lineal
3. Los archivos deben estar ordenados
4. Los archivos no deben estar ordenados
5. **Ninguna de las anteriores**

Incluyen otros temas:

34)(puede tener más de una correcta)

El NRR puede utilizarse

1. **En archivos con registro de longitud fija**
2. En archivos con registros de longitud variable
3. **Para realizar acceso directo a un registro**
4. Para realizar acceso secuencial en un archivo

Creo que hay varias que pueden ser:

35. Un archivo con 20 registros:

- a. **No necesita ordenarse**
- b. Necesita ordenarse
- c. Si se ordena, debe mantenerse ordenado
- d. Si se ordena, debe aceptar sólo búsquedas binarias
- e. Si está desordenado, la búsqueda necesita saber la posición
- f. Algunas de las anteriores
- g. Ninguna de las anteriores

36. Un algoritmo de actualización maestro detalle

fs = fun o Funciona solo con?

- a. FS detalles ordenados
- b. FS maestro ordenado
- c. FS detalles y maestro ordenados
- d. FS detalles desordenados
- e. FS maestro desordenado
- f. FS maestro y detalles desordenados
- g. Algunas de las anteriores
- h. **Ninguna de las anteriores**

37. Un algoritmo de corte de control: ¿Necesita orden?

DUDOSA, PREGUNTAR

- a. Se debe aplicar a un archivo ordenado para obtener un resultado coherente

- b. Se debe aplicar a un archivo desordenado para obtener un resultado coherente
- c. Es necesario que el archivo este ordenado
- d. Es suficiente que el archivo este ordenado por un criterio
- e. Algunas de las anteriores (a, c,¿d?)**
- f. Ninguna de las anteriores

38. Para aplicar un algoritmo de merge

- a. Es necesario más de un archivo
- b. Es necesario que el archivo maestro esté ordenado
- c. Es necesario que los detalles estén ordenados
- d. No es necesario que el archivo maestro esté ordenado
- e. No es necesario que los archivos detalles estén ordenados
- f. Algunas de las anteriores (d y e)**
- g. Ninguna de las anteriores

39. El proceso de baja lógica

- a. Se puede aplicar a archivos desordenados
- b. Se puede aplicar a archivos ordenados
- c. Se puede aplicar a archivos de longitud fija
- d. Se puede aplicar a archivos de longitud variable
- e. Se puede aplicar a cualquier archivo
- f. No se puede aplicar si el archivo está vacío
- g. Todas las anteriores**

41. Un archivo físicamente ordenado

- a. Es más fácil de recorrer
- b. Permite búsqueda binaria
- c. Permite búsqueda binaria sólo si las altas mantienen el archivo ordenado
- d. No permite búsqueda binaria si hay bajas lógicas
- e. Algunas de las anteriores (b y c)**
- f. Ninguna de las anteriores

42. Una operación de lectura en un archivo **DUDOSA, PREGUNTAR**

- a. Mueve automáticamente al siguiente registro físicamente ordenado
- b. No mueve el puntero
- c. Solo mueve si no está en el último registro
- d. Mueve automáticamente el puntero
- e. Solo se mueve el puntero después de un reset
- f. Algunas de las anteriores (a y d)**
- g. Ninguna de las anteriores

43. La función EOF

- a. Puede devolver verdadero después de un reset
- b. Puede devolver falso después de una lectura
- c. Puede devolver verdadero después de una lectura
- d. Devuelve verdadero si estas al final del archivo
- e. Devuelve falso si no estas al final del archivo
- f. Todas las anteriores**
- g. Algunas de las anteriores

h. Ninguna de las anteriores

44. La fragmentación en un archivo de longitud fija

- a. Dificulta la baja física
- b. Dificulta la baja lógica
- c. Produce pérdida de espacio**
- d. Es menor utilizando primer ajuste
- e. Algunas de las anteriores
- f. Ninguna de las anteriores

45. Con respecto a un archivo con registros de longitud fija: ¿Cuáles de estas alternativas son correctas?

- a. Utilizar NRR para tener acceso directo.**
- b. Utilizar un indicador de longitud al inicio de cada registro.
- c. Utilizar un segundo archivo con la información de la dirección del byte de inicio de cada registro.
- d. Ninguna de las anteriores.

46. El mecanismo de selección por reemplazo:

1. Reserva y utiliza un buffer en memoria secundaria para ahorrar memoria RAM.
2. Permite ordenar un archivo, aunque no entren todos sus elementos en memoria principal
3. Permite ordenar un archivo, siempre que todos sus elementos entren en memoria principal.
4. Ninguna de las anteriores.

47. La eficacia de un algoritmo de búsqueda de registros en un archivo de datos que no está ordenado:

1. Orden constante
2. **Orden Lineal**
3. Orden logarítmico
4. No tengo suficientes datos para responder
5. Ninguna de las anteriores.

48. La operación seek en un archivo:

1. Permite posicionarse directamente al final del archivo
2. Permite posicionarse directamente en el primer registro del archivo.
3. Permite conocer la posición actual dentro del archivo.
4. **Ninguna de las anteriores.**

49. Para actualizar un archivo maestro a partir de N archivos detalles:

1. Es necesario que los archivos detalles estén ordenados.
2. Es necesario que el archivo maestro esté ordenado.
3. Es necesario que todos los archivos tengan la misma estructura.
4. **Ninguna de las anteriores.**

50. Con respecto a los buffers de E/S:

1. **Ocupan lugar en memoria RAM**
2. **El SO está encargado de manipularlos.**
3. **Mejoran la performance en lecturas y escrituras.**
4. Ninguna de las anteriores.

51. El proceso de baja lógica en un archivo de datos:
repetida

1. No se puede aplicar con registros de longitud variable
2. Solo se aplica a registros de longitud variable.
3. **Permite recuperar el espacio con nuevas altas**
4. Ninguna de las anteriores.

52. La política de primer ajuste:

1. Solo se aplica en archivos con registros de longitud fija.
2. **Solo se aplica en archivos con registros de longitud variable.**
3. Puede generar fragmentación externa.
4. Ninguna de las anteriores.

53. Es posible aplicar la búsqueda binaria en:

1. Archivos desordenados con registros de longitud variable
2. Archivos desordenados con registros de longitud fija
3. Archivos ordenados con registros de longitud variable
4. **Archivos ordenados con registros de longitud fija**
5. Ninguna de las anteriores

54. Al ordenar un archivo, con respecto a la intercalación en más de un paso:

1. Consiste en agregar pasos intermedios con archivos temporales.
2. No requiere que se lea cada registro más de una vez
3. Es más eficiente que el Sort Interno.
4. Ninguna de las anteriores.