Cuando la densidad de empaquetamiento tiende a uno:

La técnica de hash extensible.

Cuáles de estas técnicas se puede utilizar con un archivo de registros de longitud fija?

En un ambiente de dispersión, cuanto más grande es el tamaño de la cubeta:

La densidad de empaquetamiento se puede definir como:

Se produce saturación: repetida

Al utilizar la técnica de dispersión doble:

La técnica de saturación progresiva encadenada:

En un ambiente de dispersión con espacio de direccionamiento estático:

Con respecto a la dispersión extensible:

Con respecto a la técnica de hash asistido por tabla:

Al usar dispersión con espacio de direccionamiento estático. ¿Cuáles de los siguientes procesos se debe realizar si se agota el espacio disponible asignado al archivo?.

Al utilizar la técnica de dispersión doble: #Repetida

Con respecto a la técnica de hash asistido por tabla:

La densidad de empaquetamiento se puede definir como:

En un ambiente de dispersión con espacio de almacenamiento estático:

Se produce saturación:

Al usar dispersión con espacio de almacenamiento estático. ¿cuales de los siguientes procesos se debe realizar si se agota el espacio disponible asignado al archivo? //PREGUNTAR

Con respecto a la dispersión extensible:

En un ambiente de dispersión, cuanto más grande es el tamaño de la cubeta:

La técnica de saturación progresiva encadenada:

¿Cual de estas técnicas se puede usar con un archivo de registros de longitud fija?:

La densidad de empaquetamiento en un archivo con registros con longitud variable:

El método de tratamiento de desborde en área separada

Indique cuál método de tratamiento de desborde potencialmente podría generar desplazamientos de disco (y consecuentemente demorar más tiempo) para encontrar un registro en saturación. Para este análisis debe suponer que el registro en saturación se aloja en la primera dirección disponible de acuerdo a la política de cada método.

El hash con espacio de direccionamiento dinámico

Implementar hash para ordenar un archivo:

El método de saturación progresiva encadenada: //PREGUNTAR

Una función de hash //PREGUNTAR

Cuales de los siguientes ítems puede considerarse atributos o propiedades de la técnica de hash

Cual de los siguientes parámetros de la técnica de hash es más importante // bajo que criterio algo es más importante que otra cosa?

Se define la densidad de empaquetamiento como

Hashing

- ***Secuencia: N/2 accesos promedio***
- ***Ordenado:*** $log_2 N$
- ***Árboles: 3 o 4 accesos***

Definiciones

- Técnica para generar una dirección base única para una llave dada. La dispersión se usa cuando se requiere acceso rápido a una llave
- Técnica que convierte la llave del registro en un número aleatorio, el que sirve después para determinar dónde se almacena el registro.
- Técnica de almacenamiento y recuperación que usa una función de hash para mapear registros en dirección de almacenamiento

Clave primaria → Características

- No se repiten
- El resto de las claves actúan a través de ella

Archivos secuenciales indexados

- Archivos de datos
- Archivo con índice primario
- Archivos con índices unívocos o secundarios

Archivos directos

- Un acceso
- No puede haber estructuras adicionales
- Se organiza el archivo de datos
- Solo puede organizarse por un único criterio
- Ese criterio es la clave primaria

Atributos del hash

- No requiere almacenamiento adicional (índice)
- Facilita insercion y eliminacion rapida de registros
- Encuentra registros con muy pocos accesos al disco en promedio

Costo

★ No podemos usar registros de longitud variable
★ No puede haber orden físico de datos
★ No permite llaves duplicadas

Para determinar la dirección

❖ La clave se convierte en un número casi aleatorio
❖ # se convierte en una dirección de memoria
❖ El registro se guarda en esa dirección
❖ Si la dirección está ocupada → Colisión / Overflow (tratamiento especial)

Eficiencia

→Función de hash:Técnica para generar la dirección
→Tamaño de los nodos:Dependiendo del disco
→Densidad de empaquetamiento:DE = n° de reg/total archivo
→Método de tratamiento de desbordes

Función de hash

- Caja negra que a partir de una clave se obtiene la dirección donde debe estar el registro
- Diferencias con índices
  - ❖ Dispersion, no hay relación aparente entre llave y direccion
  - ❖ 2 llaves distintas pueden transformarse en iguales direcciones (colisiones)

Colisión

Situación en la que el registro es asignado a una dirección que está utilizada por otro registro

Overflow

Situación en la que un registro es asignado a una dirección que está utilizada por otro registro y no queda espacio para este nuevo

Soluciones:

- Algoritmos de dispersión sin colisiones o que estas colisiones nunca produzcan overflow (perfectos)(imposibles de conseguir)
- Almacenar los registros de alguna otra forma, esparcir

Soluciones para las colisiones

- Esparcir registros: buscar métodos que distribuyan los registros de la forma más aleatoria posible
- Usar memoria adicional: distribuir pocos registros en muchas direcciones, baja la densidad de empaquetamiento
  - ❖ Disminuye las colisiones y por ende overflow
  - ❖ Desperdicia espacio
- Colocar más de un registro por dirección: direcciones con N claves, mejoras notables

Algoritmos simples de dispersión

- Condiciones
  - ❖ Repartir registros en forma uniforme
  - ❖ Aleatoria (las claves son independientes, no influyen una sobre la otra)

  3 pasos:
  - Representar la llave en forma numérica (en caso que no lo sea)
  - Aplicar la función

- Relacionar el número resultante con el espacio disponible

Tamaño de las cubetas

- Puede tener más de un registro
- A mayor tamaño
  - ❖ Menor overflow
  - ❖ Mayor fragmentación
  - ❖ Búsqueda más lenta dentro de la cubeta

Densidad de empaquetamiento

- Proporción de espacio del archivo asignado que en realidad almacena registros
- DE = n° de registros del archivo/ capacidad total del archivo
- Densidad de empaquetamiento menor
  - ❖ Menos overflow
  - ❖ Más desperdicio de espacio

Estimación del overflow

- Es necesario analizar el comportamiento de un archivo directo
- Cuando encontrar un registro requiere un solo acceso y cuando requiere más cantidad de accesos
- Estimar el overflow
  - ❖ Analizar probabilísticamente si la inserción de un registro genera o no colisión
  - ❖ Analizar si la colisión genera o no overflow
- Es necesario
  - ❖ Conocer elementos básicos de probabilidades
  - ❖ Vamos a utilizar la distribución de Poisson

Sabiendo que:

- N → Número de cubetas
- C → Capacidad de nodo
- R → Número de registro del archivo

- DE = R/C*N

Probabilidad que una cubeta reciba I registros (distribución de Poisson)

$$P(I) = \frac{R!}{I!(R-I)} * \left(\frac{1}{N}\right)^{I} * \left(1 - \frac{1}{N}\right)^{R-I}$$

Análisis numéricos de Hashing

- En general, si hay n direcciones, entonces el número esperado de direcciones con I registros asignados es N * P(I)
- Las colisiones aumentan con el archivo más "lleno"

Tratamiento de colisiones con overflow

Hemos visto que el % de overflow se reduce, pero el problema se mantiene dado que no llegamos a 0%

Algunos métodos

- Saturación progresiva
- Saturación progresiva encadenada
- Doble dispersión
- Área de desborde separada

Saturación progresiva

- Cuando se completa el nodo, se busca el proximo hasta encontrar uno libre
- Búsqueda ?
- Eliminación, no debe obstaculizar las búsquedas

Saturación progresiva encadenada

- Similar a saturación progresiva, pero los registros de saturación se encadenan y "no ocupan" necesariamente posiciones contiguas

- Se hace un puntero al próximo nodo que tiene el dato de esa dirección

Dispersión doble

- Saturación tiende a agrupar en zonas contiguas, búsquedas largas cuando la densidad tiende a uno
- Solución: Almacenar los registros de overflow en zonas no relacionadas
- Esquema con el cual se resuelven overflows aplicando una 2da función a la llave para producir un número C, el cual se suma a la dirección original tantas veces como sea necesario hasta encontrar una dirección con espacio

Encadenamiento con áreas separadas

- No utiliza nodos de direcciones para los overflow, estos van a nodos especiales
- Se reservan X nodos para overflow, cuando en una casilla se produce overflow, voy al lugar del espacio reservado y lo guardo ahí

Hash con espacio de direccionamiento estático

- Necesita un número de direcciones fijas, virtualmente imposible
- Cuando el archivo se llena
  - ❖ Saturación excesiva
  - ❖ Redispersar, nueva función, muchos cambios

Solución → espacio de direccionamiento dinámico

- Reorganizar tablas sin mover muchos registros
- Técnicas que asumen bloques físicos, pueden utilizarse o liberarse

Espacio dinámico

- Hash virtual
- Hash dinámico
- Hash extensible

Hash extensible

- Adapta el resultado de la función de hash de acuerdo al número de registros que tenga el archivo, y de las cubetas necesitadas para su almacenamiento
- Función: Genera secuencias de bits (normalmente 32)

¿Cómo trabaja?

- Se utilizan solo los bits necesarios de acuerdo a cada instancia del archivo
- Los bits tomados forman la dirección del nodo que se utilizará
- Si se intenta insertar a una cubeta llena, deben reubicarse todos los registros allí contenidos entre el nodo viejo y el nuevo, para ello se toma un bit más
- La tabla tendrá tantas entradas (direcciones de nodos) como 2^i, siendo i el número de bits actuales para el sistema

Viola una de las propiedades del hash

Correcciones parciales:

Para cuando se debe poner marca de borrado, tiene que pasar: Encuentro el elemento en X nodo, dicho nodo debe estar lleno y la posición siguiente debe tener algún elemento.
En árboles, cuando se intercambian elementos, no se debe escribir en el momento del intercambio, sino que se hace una única escritura cuando se terminan las operaciones. Y los nodos liberados, no toman como escritura, sino que se debe tener un listado de nodos liberados.

Preguntas

1. Cuál de las siguientes definiciones corresponden al método de hash:

    1. Técnica para generar una dirección base única para una clave dada
    2. Técnica que convierte la clave asociada a un registro de datos en un número aleatorio, que se utiliza para determinar dónde se almacena el registro
    3. Técnica de almacenamiento y recuperación que usa una función para mapear registros en direcciones de almacenamiento en memoria secundaria.
    4. **Todas las anteriores son aplicables**

2. La técnica de hash:

    1. Entorpece la inserción y borrado de elementos.
    2. La localización de un registro siempre debe utilizar una tabla adicional en memoria.
    3. **No es conveniente de aplicar sobre claves secundarias**,
    4. Requiere al menos de dos funciones de hash para el tratamiento de los desbordes.

3. Cuál de los siguientes conceptos corresponden con parámetros de la dispersión.

    1. Capacidad de almacenamiento de cada sector del archivo
    2. Densidad de Empaquetamiento
    3. Método de tratamiento de desbordes.

4. **Todos los anteriores**.

4. La densidad de empaquetamiento se define como:

1. El cociente entre cantidad de registros y espacio disponible en el archivo.
2. El cociente entre la cantidad de registros y la cantidad de nodos del archivo.
3. El cociente entre la cantidad de registros, y el producto entre la cantidad de nodos y el contenido posible de registros de cada nodo.
4. **Hay más de una respuesta correcta** (1 y 3)

5. Una colisión se produce:

1. **Cuando dos registros diferentes obtienen de la función de hash la misma dirección de disco.**
2. Cuando dos registros iguales obtienen de la función de hash direcciones diferentes de disco.
3. Cuando un registro no cabe en el lugar donde debe almacenarse de acuerdo al resultado de la función de hash.
4. Cuando dos registros diferentes obtienen de la función de hash direcciones diferentes de disco.

7. La eficiencia promedio de búsqueda en un archivo a partir de estar organizado mediante política de Hashing:

1. Orden Lineal.
2. Orden logarítmico.
3. **Orden Constante.**
4. No dispongo de datos para contestar la pregunta.

8. La eficiencia de búsqueda de un registro en un archivo organizado mediante hashing estático tiene:

    1. Orden Lineal
    2. **Algunas veces es uno**
    3. Siempre es uno
    4. Orden Logarítmico.

9. Cuando la densidad de empaquetamiento tiende a uno:

    1. Es necesario redefinir el espacio disponible únicamente.
    2. El archivo se completa y no es posible incorporar más elementos.
    3. Se debe cambiar la política de hash de estática a dinámica.
    4. **Es necesario redefinir el espacio disponible y rehashear todo el archivo.**

10. La técnica de hash extensible.

    1. Presenta una variante de hash que permite no sólo ubicar rápidamente los registros sino que además permite el acceso secuencial a los mismos.
    2. Siempre inserta un registro con un y solo un acceso a disco.
    3. **Siempre se recupera un registro con un y sólo un acceso a disco.**
    4. En algunos casos recupera un registro con un y solo un acceso a disco.

(De la 11 a la 20 hay más de una correcta)

11. ¿Cuáles de estas técnicas se puede utilizar con un archivo de registros de longitud fija?

    1. **Dispersión doble.**
    2. **Hashing asistido por tabla.**
    3. **Hashing extensible.**
    4. Ninguna de las anteriores.

12. En un ambiente de dispersión, cuanto más grande es el tamaño de la cubeta:

    1. **Hay más fragmentación.**
    2. Hay mayor probabilidad de saturación.
    3. **La búsqueda dentro de la cubeta es más lenta.**
    4. Ninguna de las anteriores

13. La densidad de empaquetamiento se puede definir como:

    1. La relación entre la cantidad de registros y la cantidad de cubetas del archivo.
    2. **La proporción de espacio asignado al archivo que en realidad almacena registros.**
    3. DE = cantidad de registros / (cantidad de cubetas)

4. Ninguna de las anteriores

14. Se produce saturación

1. Siempre que dos registros diferentes obtienen de la función de hash la misma dirección de disco.

2. Siempre que dos registros iguales obtienen de la función de hash direcciones diferentes de disco.

3. **Cuando un registro no cabe en el lugar donde debe almacenarse según el resultado de la función de hash.**

4. Cuando dos registros diferentes obtienen de la función de hash direcciones diferentes de disco.

5. Ninguna de las anteriores.

15. Al utilizar la técnica de dispersión doble:

1. Dada una clave, siempre se debe aplicar dos funciones: inicialmente se aplica la 1ra función,y al resultado se le aplica la 2da función para obtener finalmente la dirección de almacenamiento.

2. **Se cuenta con dos funciones, pero sólo se aplica la 2da función si se produjo una saturación al aplicar la 1ra función.**

3. Cuando se usa la 2da función, el valor obtenido reemplaza al anteriormente obtenido por la 1ra función.

4. Ninguna de las anteriores.

16. La técnica de saturación progresiva encadenada:

1. Evita la generación de colisiones.

2. Necesita que cada cubeta tenga capacidad para dos o más registros.
3. Requiere al menos de dos funciones de hash para el tratamiento de los desbordes.
4. **Ninguna de las anteriores**.

17. En un ambiente de dispersión con espacio de direccionamiento estático:

1. Siempre se encuentra un registro con un solo acceso a disco.
2. **No se permiten claves duplicadas.**
3. Es posible usar archivos con registros de longitud variable.
4. Ninguna de las anteriores

18. Con respecto a la dispersión extensible:

1. **El espacio aumenta o disminuye dependiendo de los registros que contiene el archivo.**
2. **Necesita una tabla auxiliar.**
3. La densidad de empaquetamiento siempre se mantiene por debajo del 75%.
4. Ninguna de las anteriores.

20. Al usar dispersión con espacio de direccionamiento estático. ¿Cuáles de los siguientes procesos se debe realizar si se agota el espacio disponible asignado al archivo?

1. Iniciar un nuevo archivo y relacionarlo con el archivo que quedó completo.
2. **Obtener más espacio para el mismo archivo, actualizar la función de hash, y redispersar el archivo completo.**
3. Obtener más espacio para el mismo archivo, actualizar la función de hash pero no redispersar (se usa la nueva función sólo para los nuevos elementos).
4. No es posible que se agote el espacio disponible asignado al archivo.
5. Ninguna de las anteriores.

23. La densidad de empaquetamiento se puede definir como:

1. **DE = cantidad de registros / (cantidad de cubetas * capacidad de cubeta).**
2. **La proporción de espacio asignado al archivo que en realidad almacena registros.**
3. La relación entre la cantidad de registros y la cantidad de cubetas del archivo.
4. Ninguna de las anteriores.

24. En un ambiente de dispersión con espacio de almacenamiento estático:

1. **Se debe usar archivos con registro de long fija.**

2. Siempre se encuentra un registro con un solo acceso a disco.

3. **No existe orden físico de datos.**

4. Ninguna de las anteriores.

25. Se produce saturación:

1. Siempre que dos registros diferentes obtienen de la función de hash la misma dirección de disco.

2. Siempre que dos registros iguales obtienen de la función de hash direcciones diferentes de disco.

3. Cuando dos registros diferentes obtienen de la función de hash direcciones diferentes de disco

4. **Ninguna de las anteriores.**

26. Al usar dispersión con espacio de almacenamiento estático. ¿cuales de los siguientes procesos se debe realizar si se agota el espacio disponible asignado al archivo? //**PREGUNTAR**

1. Iniciar un nuevo archivo y relacionarlo con el archivo que quedó completo.

2. Obtener más espacio para el mismo archivo y actualizar la función de hash no es necesario dispersar el archivo , ya que se usa la nueva función solo para los nuevos elementos.

3. No es posible que se agote el espacio asignado al archivo.

4. **Ninguna de las anteriores.**

29. La técnica de saturación progresiva encadenada:

1. **Es posible aplicarla, aunque cada cubeta tenga capacidad para un solo registro.**
2. Evita la generación de colisiones.
3. Requiere de al menos dos funciones de hash para el tratamiento de los desbordes.
4. Ninguna de las anteriores.

(Solo una correcta a partir de aca)

31. La densidad de empaquetamiento en un archivo con registros con longitud variable:

1. Se calcula como el cociente entre la cantidad de registros del archivo y la cantidad de espacio disponible.
2. Es útil para establecer la proporción de espacio del archivo asignado que en realidad almacena registros.
3. A medida que disminuye, aumenta la probabilidad de overflow.
4. A medida que disminuye, hay más desperdicio de espacio.
5. Alguna de las anteriores.
6. **Ninguna de las anteriores.**

32. El método de tratamiento de desborde en área separada

1. Utiliza una segunda función de hash para determinar donde va el registro en overflow.
2. Utiliza una segunda función de hash para determinar el nuevo nodo donde se guardará el registro en overflow.

3. Selecciona el primer nodo libre más cercano al nodo saturado
4. Selecciona el primer nodo libre más cercano al saturado y los linkea
5. **Ninguna de las anteriores.**

33. Indique cuál método de tratamiento de desborde potencialmente podría generar desplazamientos de disco (y consecuentemente demorar más tiempo) para encontrar un registro en saturación. Para este análisis debe suponer que el registro en saturación se aloja en la primera dirección disponible de acuerdo a la política de cada método.

1. Saturación progresiva
2. Saturación progresiva encadenada.
3. Doble dispersión
4. **Área de desborde separada**

34. El hash con espacio de direccionamiento dinámico

1. Utiliza una función de hash diferente cada vez que el tamaño del archivo crece
2. Aumenta la capacidad del nodo cuando se puede producir un overflow
3. **Aumenta la cantidad de nodos en caso de overflow**
4. Todas la anteriores
5. Ninguna de las anteriores.

35. Implementar hash para ordenar un archivo:

1. Mejora el acceso a un archivo.
2. Puede empeorar el acceso a los datos de un archivo.
3. Si es hash extensible la mejora es notable.
4. **El concepto no es aplicable.**

36. El método de saturación progresiva encadenada: //PREGUNTAR

1. Es más eficiente que doble dispersión porque utiliza una sola función de hash
2. Se demora siempre menos en la búsqueda que utilizando el método de área separado
3. **Es un método más de tratamiento de colisiones**
4. Siempre es más eficiente que cualquier situación similar resultante con el método de saturación progresiva.

37. Una función de hash //PREGUNTAR

1. Siempre devuelve una dirección donde se debería almacenar el registro dispersado
2. Siempre devuelve una secuencia de bits que sirve para determinar dónde se debería almacenar el registro dispersado.
3. **Siempre debe ser aleatoria.**
4. Todas las anteriores.

5. Algunas de las anteriores.

38. Cuáles de los siguientes ítems puede considerarse atributos o propiedades de la técnica de hash

1. No requerir espacio adicional de almacenamiento
2. Facilitar inserción y eliminación rápida de elementos en el archivo
3. Permitir la búsqueda más eficiente de información en un archivo de datos
4. Permitir acceso secuencial a los datos
5. Todas las anteriores
6. **Algunas de las anteriores.** (2, 3 y 1)

39. Cual de los siguientes parámetros de la técnica de hash es más importante // bajo que criterio algo es más importante que otra cosa?

1. Función de hash
2. Tamaño de cada nodo de almacenamiento
3. Densidad de empaquetamiento
4. Método de tratamiento de overflow
5. Ninguno es importante
6. **Ninguno prevalece sobre el otro**.

40. Se define la densidad de empaquetamiento como

1. El cociente entre la cantidad de registros del archivo y el espacio disponible para su almacenamiento

2. El cociente entre la capacidad del archivo y los registros del archivo

3. El cociente entre la cantidad de registros del archivo, y el producto entre la cantidad de nodos disponibles y la capacidad de estos nodos.

4. Las respuestas a,b y c son correctas.

5. Las respuestas a y b son correctas.

6. **Las respuestas a y c son correctas.**

7. Las respuestas b y c son correctas.

8. No hay respuesta correcta