

Ordenamiento Externo

Unidad 6.

Almacenamiento y Estructuras de Archivos

Ordenamiento interno

Introducción

- El ordenamiento consiste en ordenar los registros de un archivo homogéneo basándose en el valor de alguna llave y puede ser descendente o ascendente
- El ordenamiento es una de las aplicaciones más utilizadas en la computación
- La mayoría de los datos están ordenados de alguna manera
- Muchos de los cálculos invocan internamente a un método de ordenamiento

Algoritmos

- El ordenamiento interno consta de tres pasos en general:
 - Leer los datos
 - Ordenar los datos en memoria
 - Escribir o presentar los datos
- Existen diversos algoritmos de ordenamiento:
 - Bubblesort
 - Mergesort
 - Quicksort
 - Ordenamiento por inserción
 - Shellsort

Bubblesort

- Es el más sencillo de los algoritmos de ordenamiento
- Revisa cada elemento de la lista que va a ser ordenada con el siguiente, intercambiándolos de posición si están en el orden equivocado
- Es necesario revisar varias veces toda la lista hasta que no se necesiten más intercambios
- Este algoritmo obtiene su nombre de la forma con la que suben por la lista los elementos durante los intercambios, como si fueran "burbujas"

Ejemplo

4,3,5,2,1	3,4,5,2,1
3,4,5,2,1	3,4,5,2,1
3,4,5,2,1	3,4,2,5,1
3,4,2,5,1	3,4,2,1,5
3,4,2,1,5	3,4,2,1,5
3,4,2,1,5	3,2,4,1,5
3,2,4,1,5	3,2,1,4,5
3,2,1,4,5	3,2,1,4,5

3,2,1,4,5	2,3,1,4,5
2,3,1,4,5	2,1,3,4,5
2,1,3,4,5	2,1,3,4,5
2,1,3,4,5	2,1,3,4,5
2,1,3,4,5	1,2,3,4,5
1,2,3,4,5	1,2,3,4,5
1,2,3,4,5	1,2,3,4,5
1,2,3,4,5	1,2,3,4,5

Quicksort

- Es el algoritmo de ordenación más rápido
- El algoritmo *quicksort* es recursivo y está formado por cuatro pasos:
 1. Dado un conjunto S de números, si el número de elementos es menor o igual a 1, terminar.
 2. Escoger un elemento cualquiera de S , el cuál será llamado pivote $\{v\}$
 3. Hacer una partición de $S - \{v\}$ en dos grupos disjuntos:
 - $I = \{x \in S - \{v\} \mid x < v\}$
 - $D = \{x \in S - \{v\} \mid x > v\}$
 4. Devolver el resultado de *Quicksort* (I), seguido de v y luego el resultado de *Quicksort* (D)

Selección del pivote

- El algoritmo básico de Q_{sort} permite tomar cualquier elemento como pivote
- De la elección del pivote depende la eficiencia del algoritmo
- Algunos métodos para elegir el elemento que se utilizará como pivote son:
 - Elección aleatoria del pivote
 - No se requiere ningún cálculo adicional
 - El algoritmo puede tener un gran número de iteraciones
 - Recorrer la lista para ver que pivote es el conveniente
 - Tomar el elemento de que está en medio del arreglo

Ejemplo

13,81,92,43, 65 ,31,57,26,75,0
13,43, 31 ,26,0, 65, 81,92,75
13, 26 ,0, 31, 43, 65 ,81,92,75
13,0, 26, 31, 43, 65, 81,92,75
0,13,26,31,43,65, 81, 92 ,75
0,13,26,31,43,65, 81,75, 92
0,13,26,31,43,65,81,75,92

Mergesort

- La idea de mergesort es la misma para todos los métodos de ordenamiento externo
- Ordenar, mediante algún algoritmo de ordenamiento interno, una cantidad de datos que pueda almacenarse en memoria
- Combinar los distintos grupos de datos mediante el método de mezcla

Mergesort

- El algoritmo de mezcla parte de dos vectores de entrada A y B produciendo un vector de salida C
- Se apoya en tres contadores
 - Acont
 - Bcont
 - Ccont

Mergesort

- En cada paso el menor de los valores A y B se copia en la próxima posición de C, actualizándose los contadores apropiados
- Cuando uno de los vectores de entrada se acaba, el resto del otro vector se copia en C

Ejemplo

1,13,24,26 2,15,24,38	
1,13,24,26 2,15,24,38	1
1,13,24,26 2,15,24,38	1,2
1,13,24,26 2,15,24,38	1,2,13
1,13,24,26 2,15,24,38	1,2,13,15
1,13,24,26 2,15,24,38	1,2,13,15,24
1,13,24,26 2,15,24,38	1,2,13,15,24,24
1,13,24,26 2,15,24,38	1,2,13,15,24,24,26
1,13,24,26 2,15,24,38	1,2,13,15,24,24,26,38

Ordenamiento externo

Introducción

- Cuando la cantidad de datos a ordenar es tan grande que no cabe en memoria se debe recurrir al ordenamiento externo
- Los métodos mas comunes utilizan un proceso de intercalación
- La intercalación de dos archivos es una operación que toma como entrada dos o más archivos homogéneos que ya están ordenados en base a la misma llave y genera un nuevo archivo que contiene los los registros de los archivos de entrada ordenados

Características

- Las principales propiedades del ordenamiento externo son:
 - Ordena archivos de cualquier tamaño
 - La lectura de la entrada es secuencial
 - La escritura a la salida es secuencial

Intercalación

- Pre condiciones
 - Los archivos de entrada contienen registros homogéneos
 - Los archivos de entrada deben estar ordenados en base a la misma llave
- Post condiciones
 - El archivo de salida quedará ordenado por la misma llave que los archivos de entrada
 - El archivo de salida contendrá todos los registros de los archivos de entrada

Mezcla directa

Mezcla directa

- Suponer que 900 MB de información deben ser ordenados utilizando únicamente 100 MB de RAM
 - Leer 100MB de información en memoria principal y ordenar utilizando un algoritmo tradicional (normalmente *QSort*)
 - Escribir la información ordenada en disco
 - Repetir hasta que toda la información esté ordenada en pedazos de 100 MB

Mezcla directa

- Mezclar todos los pedazos ordenados:
 - Leer los primeros 10MB de cada pedazo ordenado a la memoria principal (total de 90 MB) y dejar 10 MB restantes para el buffer de salida
 - Buscar el menor de los nueve pedazos mezclándolos y grabar el resultado en el buffer de salida
 - El pedazo de los 9 buffers leídos que quedó vacío, se llena con los siguientes 10 MB de su pedazo original de 100 MB o se marca como completado si ya no se tienen más registros

Ejemplo

- Ordenar un archivo con la siguiente información:
4,18,20,77,50,2,61,69,36,80,13,79,19,4,43,21,6,75,6,3
- Suponer que cada dato ocupa 1MB de memoria y que la computadora solo puede manejar 5MB

Árbol de ganadores

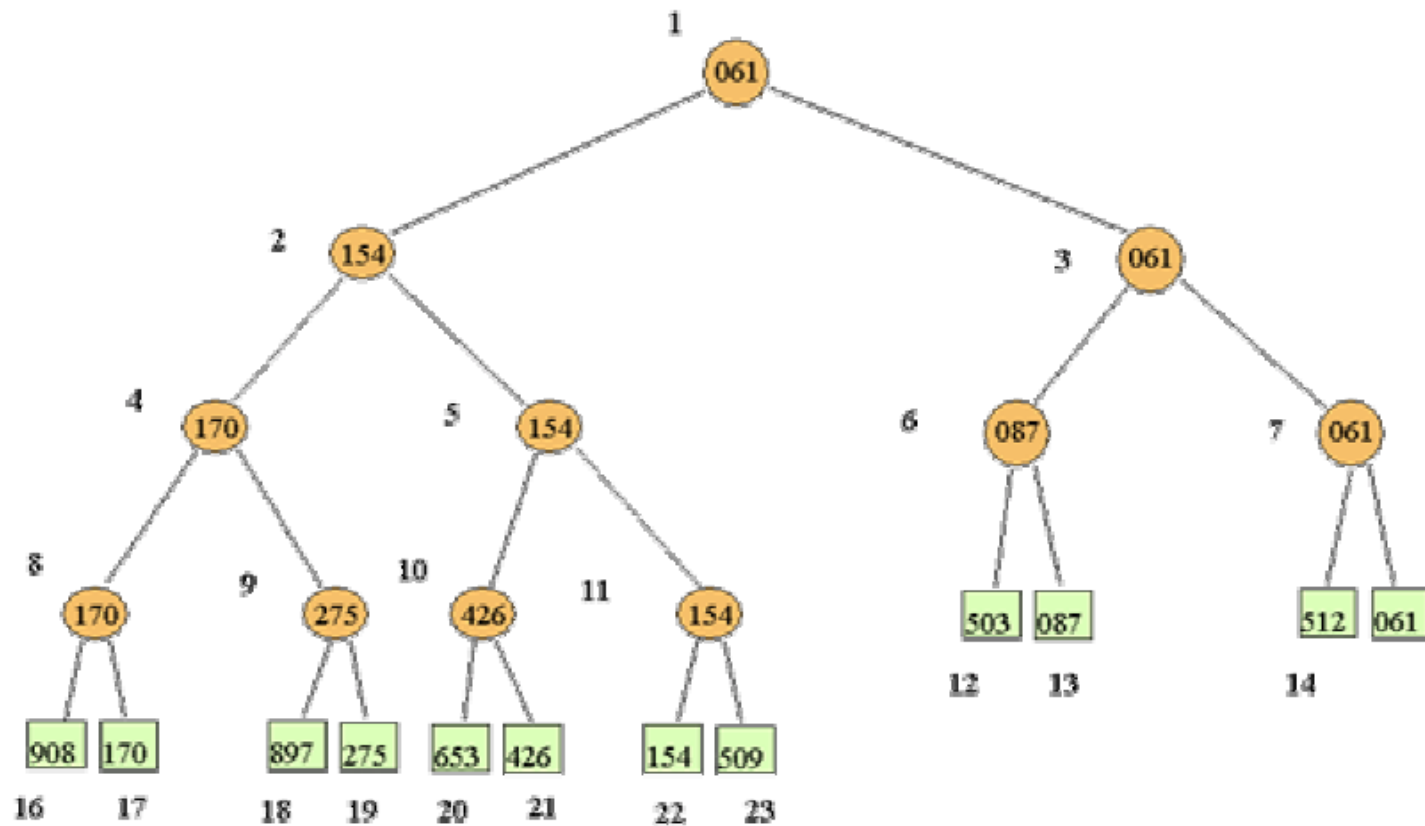
Introducción

- Una manera de hacer más eficiente el método de mezcla directa es utilizando árboles de ganadores
- El principio de este algoritmo, consiste en tener en memoria un conjunto de registros y seleccionar el que tenga el menor valor de acuerdo a una determinada llave
- Una vez escrito se debe sustituir con un nuevo registro

Funcionamiento

- Para determinar cuál es el menor de los elementos, en lugar de realizar una comparación entre los k registros en memoria ($k-1$ comparaciones) se puede utilizar un árbol de ganadores (árbol de prioridad) que reduce el costo a $\log_2(k)$ comparaciones

Ejemplo



Descripción

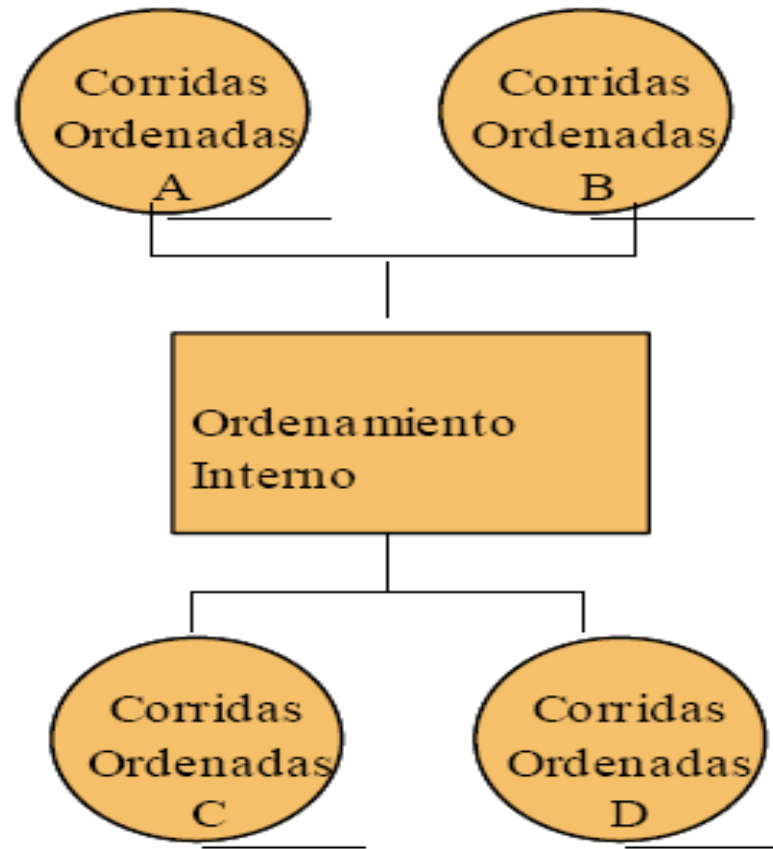
- En la figura, las hojas contienen los registros de los archivos y en los niveles superiores se registran los ganadores de las comparaciones
- En el ejemplo se tiene la llave 61, la cuál compitió con las llaves 512, 87 y 154
- Cuando se inserta la llave 61 en el archivo de salida se inserta un nuevo registro del archivo al que ésta llave pertenecía, siendo necesario solo comparar con tres llaves para encontrar al nuevo ganador

Mezcla equilibrada

Mezcla equilibrada

- Es una optimización de la mezcla directa
- Se basa en crear archivos con secuencias ordenadas de registros y posteriormente una serie de pasos de intercalación
- Se realiza la mezcla de secuencias ordenadas, alternadamente sobre otros archivos. Repitiendo este proceso secuencialmente, se logra que el archivo quede completamente ordenado
- Es muy utilizada para ordenar el contenido de cintas e donde solo puede usarse el acceso secuencial

Mezcla equilibrada de dos vías

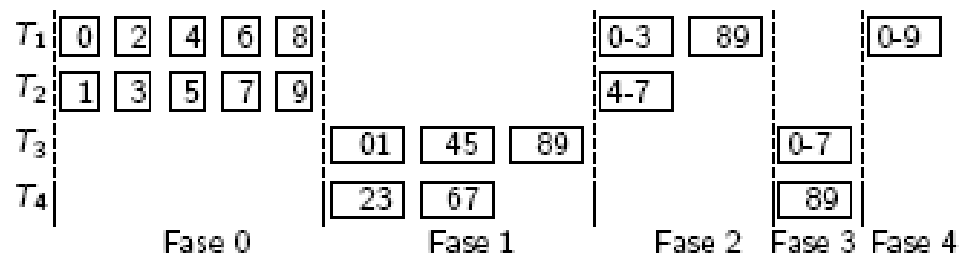


Descripción

- Se intercala la primera corrida de A con la primera de B y se graba el resultado en C
- La segunda corrida de A se intercala con la segunda de B y se graba el resultado en D
- Se repite la operación alternando las salidas en C y D hasta que se recorren completamente A y B
- Posteriormente se repite el proceso utilizando ahora a C y D como entrada y como salida se utilizan A y B
- El proceso termina cuando se encuentran todos los registros ordenados

Consideraciones

- Es necesario que las corridas estén distribuidas en dos cintas
- A cada paso de la mezcla (con excepción del último) la salida estará distribuida en dos cintas
- Son necesarias cuatro cintas para realizar el ordenamiento
- Si se tienen n corridas, el número de fases requeridas es igual a $\log_2(n)$



Mezcla equilibrada de n vías

- Método similar al anterior, solo que ahora se utilizan n cintas de entrada y n cintas de salida
- Para ordenar n corridas con k vías se necesitan $\log_k(n)$

