
Estructura de datos y Algoritmos

Tema III

Clasificación en memoria secundaria

3.1. Clasificación externa basada en mezcla

3.1.1. Mezcla directa.

3.1.2. Mezcla natural.

3.1.3. Mezcla balanceada múltiple.

3.1.4. Clasificación polifásica.

3.2. Archivos Indexados

3.3 Tablas de dispersión (Hashing)

3.1. Clasificación externa basada en mezcla

- Problema: Los datos a ordenar no caben todos en memoria principal , están almacenados en dispositivos periféricos de acceso secuencial:
 - Cinta: TDA secuencia
 - Disco: TDA archivo
 - Restricciones en el proceso de clasificación:
 - Acceso secuencial a cada uno de los elementos de una secuencia.
 - No permite aplicar los métodos de clasificación sobre arreglos en los que el acceso a cualquier elemento implicaba el mismo coste.
-

Tipos de técnicas de clasificación externa basadas en Mezcla:

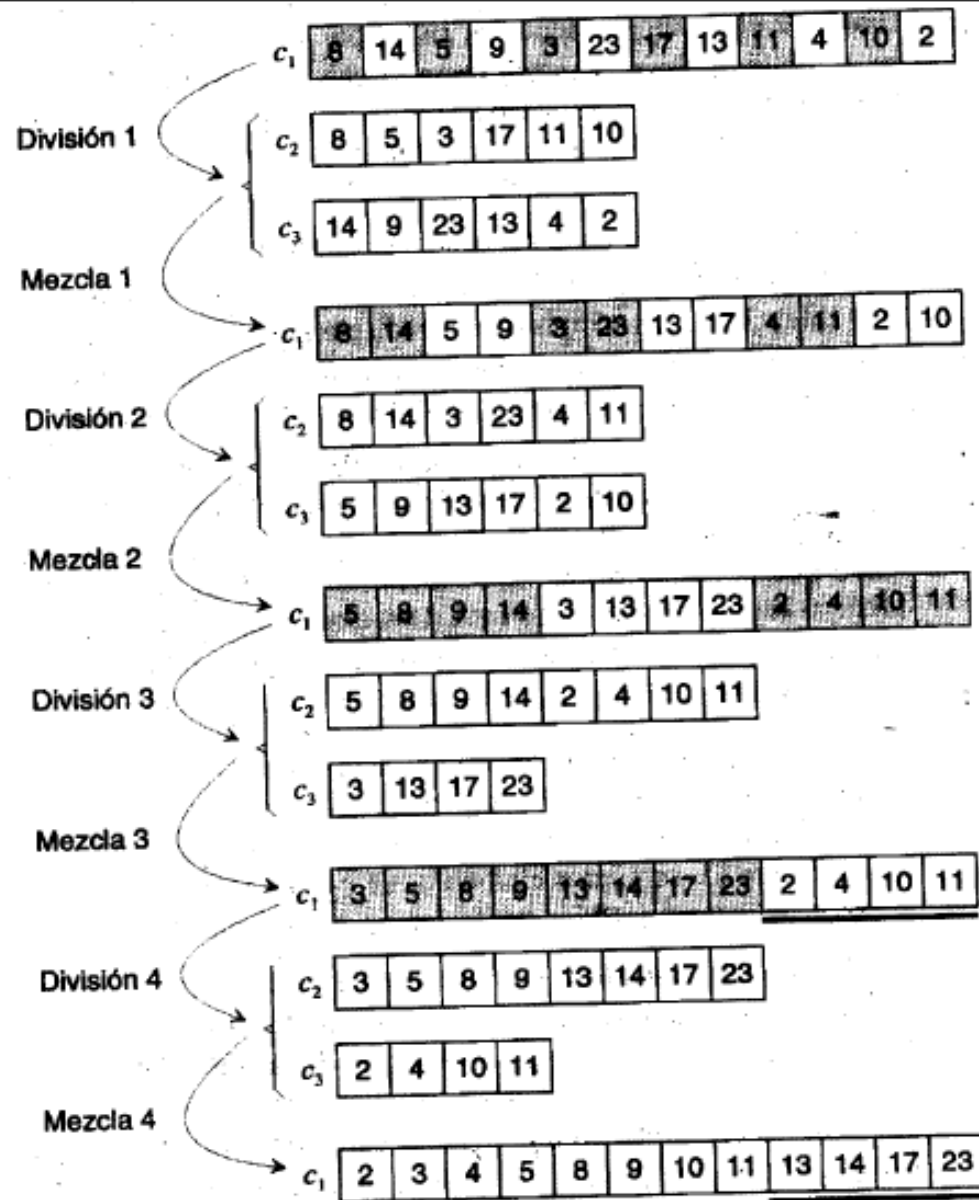
- Mezcla directa
 - Mezcla directa de una fase
 - Mezcla natural
 - Mezcla balanceada múltiple
 - Clasificación polifásica
-

Definiciones:

- Mezcla:
 - Tarea de combinar varias secuencias en una sola, mediante la selección repetida de los componentes accesibles en cada momento.
 - Es una operación auxiliar previa utilizada como estrategia para desarrollar la tarea de clasificación.
 - Fase:
 - Cada operación que trata un conjunto completo de datos. Ejemplos: distribución, mezcla.
 - Pase (etapa):
 - El proceso más corto que, por repetición, forma parte del proceso de clasificación. Ejemplo: distribución + mezcla.
 - Cinta: cada una de las secuencias necesarias en el proceso de clasificación.
-

3.1.1. Mezcla directa

- Tomar como fuente la secuencia original c_1
 - Dividir la fuente en dos mitades, en las cintas destino c_2 y c_3
 - Mezclar c_2 y c_3 combinando cada elemento accesible en pares ordenados, en c_1
 - Repetir el proceso:
 - Se obtiene una cinta con cuádruplos ordenados.
 - Repetir el proceso hasta que toda la cinta esté ordenada
 - Cada pase o etapa consta de dos fases,
 - Una de **división** y
 - Otra de **mezcla**, por ello se denomina mezcla de dos fases o mezcla de tres cintas.
-



mplo de mezcla directa. La barra bajo algunos elementos indica el copiado de cabos.

Fase de división

```
PROCEDURE Separa(VAR c,a,b: File;p: CARDINAL;
VAR longc: CARDINAL);
(*Divide c en a y b en grupos de p elementos (1,2,4,8...) y
en longc devuelve la longitud de c*)
VAR
  i,j: CARDINAL;

BEGIN
  Reset(c);
  Create(a,'nuevoa.dat');
  Create(b,'nuevob.dat');
  i:=0; j:=0; longc:=0;
  ReadWord(c,w);
  WHILE ~c.eof DO
    WHILE (~c.eof) & (i<p) DO
      WriteWord(a,w);
      ReadWord(c,w);
      i:=i+1;
      longc:=longc+1;
    END;
    i:=0;
    WHILE (~c.eof)& (j<p) DO
      WriteWord(b,w);
      ReadWord(c,w);
      j:=j+1;
      longc:=longc+1;
    END;
    j:=0;
  END;
END Separa;
```


Fase de Mezcla

```
PROCEDURE mezcla(VAR a, b, c: File; p, longc: CARDINAL);
(*Mezcla a y b en c en grupos ordenados de 2p elementos*)
VAR q, (*n° elementos que quedan por mezclar de la 1ª subsec*)
    r, (*n° elementos que quedan por mezclar de la 2ª subsec*)
    m: CARDINAL; (*n° de elementos por mezclar*)
    wa, wb: INTEGER;
BEGIN
  Reset(a); Reset(b); Reset(c); m:=longc;
  REPEAT
    IF m>=p THEN q:=p ELSE q:=m END;
    m:=m-q;
    IF m>=p THEN r:=p ELSE r:=m END;
    m:=m-r;
    (* Proposicion de mezcla *)
    ReadWord(a,wa); ReadWord(b,wb);
    WHILE (q#0)&(r#0) DO
      IF wa<wb THEN
        WriteWord(c,wa); q:=q-1;
        IF q#0 THEN ReadWord(a,wa); END
      ELSE
        WriteWord(c,wb); r:=r-1;
        IF r#0 THEN ReadWord(b,wb); END
      END
    END;
    (* Copia de cabos *)
    WHILE r>0 DO
      WriteWord(c,wb); r:=r-1;
      IF r#0 THEN ReadWord(b,wb); END
    END;
    WHILE q>0 DO
      WriteWord(c,wa); q:=q-1;
      IF q#0 THEN ReadWord(a,wa); END
    END;
  UNTIL m=0;
END mezcla;
```

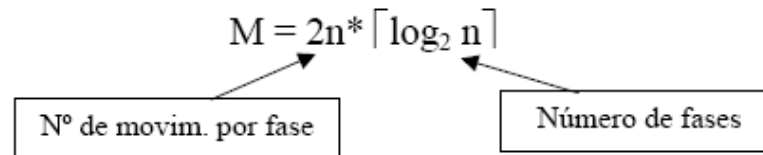
Clasificación por Mezcla Directa (Programa principal)

```
MODULE Directa;
FROM InOut IMPORT WriteInt, WriteString, WriteLn,
ReadInt;
FROM FileSystem IMPORT File,Close,
Reset,WriteWord, ReadWord, Create;

BEGIN
  p:=1;
  longc:=0; (*contador del numero de elementos de c*)
  REPEAT
    Separa(c,a,b,p,longc);
    mezcla(a,b,c,p,longc);
    p:=2*p;
  UNTIL p>=longc;
  (* Fin de ordenación *)
  Close(c);
  Close(a);
  Close(b);
END Directa.
```

Análisis de clasificación por mezcla directa

Movimientos:



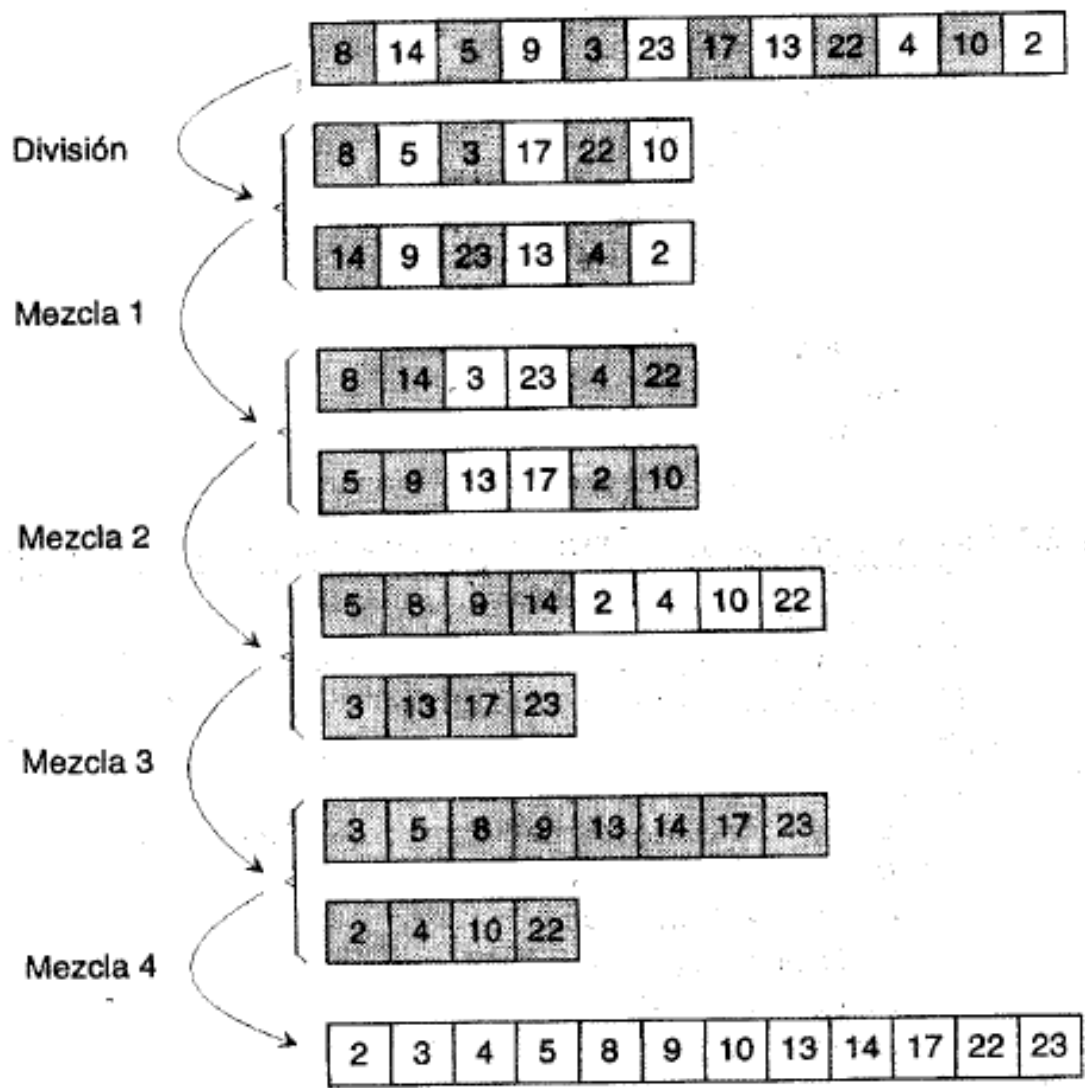
Comparaciones: menor que M, ya que no hay comparaciones en las operaciones de copiado de cabos.

$$C \leq M$$

- En las mezclas el número de comparaciones es de poco interés práctico frente a la penalización de tiempo que supone el realizar movimientos.
- La complejidad es parecida a la de los algoritmos avanzados de clasificación en memoria principal.

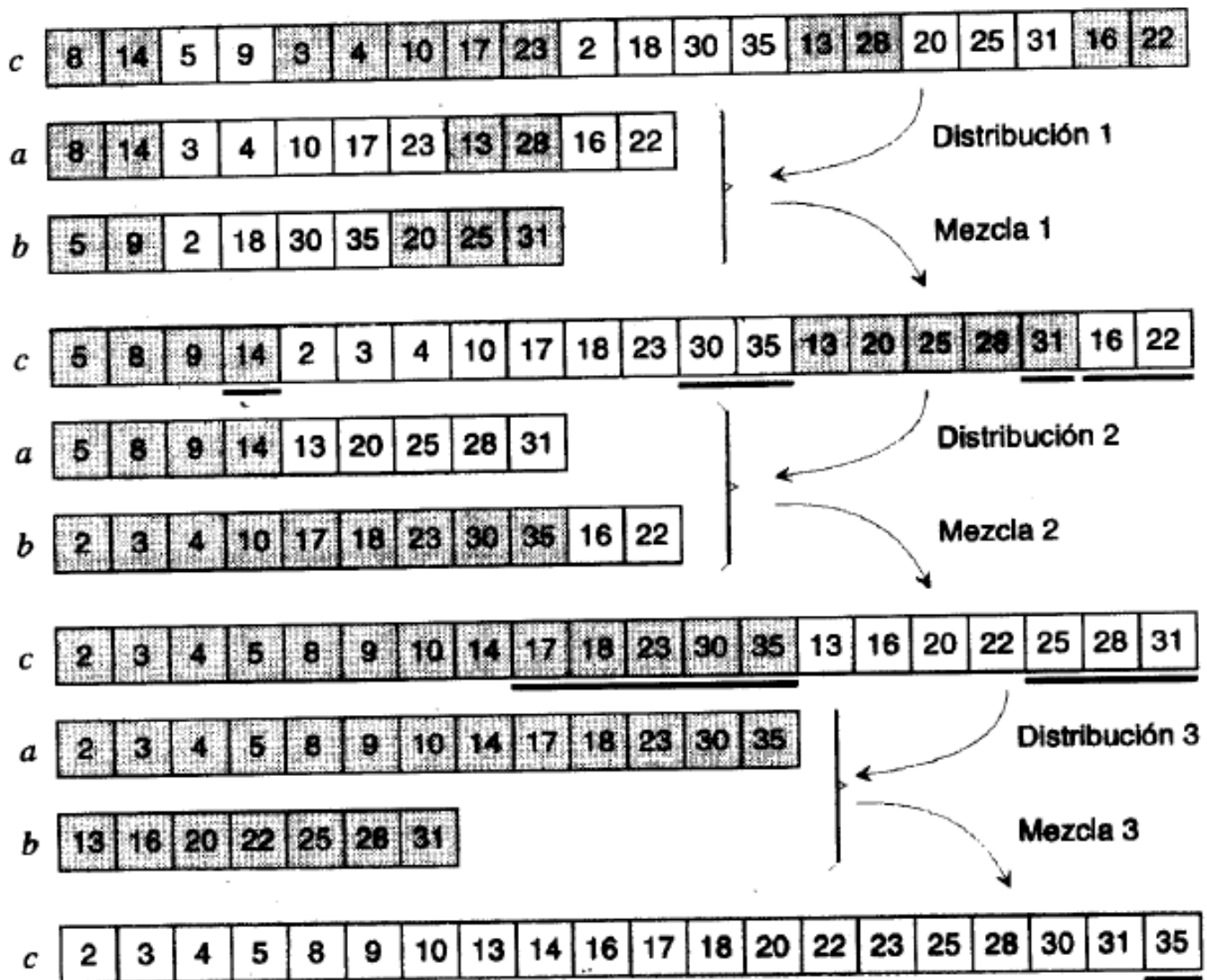
3.1.2 Mezcla Directa de una Fase (Mezcla directa balanceada)

- Debemos observar que la fase de división no aporta nada a la clasificación y sin embargo tiene un coste computacional significativo
- Las fases de división de la mezcla directa pueden eliminarse combinando la fase de división con la de mezcla
 - Mejora el método de mezcla directa
- En vez de mezclar en una sola cinta, que posteriormente será dividida, puede irse directamente separando en dos, de manera que en el siguiente pase ya estarán divididas (se elimina la distribución).
- Análisis
 - Se producen el entero mayor de $\log n$ pases. por tanto el número de movimientos es: Movimientos: $M \approx n * \lceil \log_2 n \rceil$
 - El número de comparaciones:
operaciones de copiado de cabos no se produce ninguna.



3.1.2. Mezcla natural

- Aprovecha el hecho de que entre los elementos de la secuencia original, algunos elementos consecutivos ya se encontrarán ordenados entre sí.
 - La mezcla natural se basa en la combinación de subsecuencias ordenadas. Las subsecuencias ordenadas de la cinta fuente, se distribuyen en dos cintas destino auxiliares a y b. Seguidamente se mezcla una subsecuencia ordenada de cada cinta auxiliar.
 - Fuente c
 - Distribuir las subsecuencias ordenadas de la fuente en las cintas destino a y b
 - Mezclar a y b en c, combinando subsecuencias ordenadas de cada cinta auxiliar.
 - Cada pase en la mezcla natural consta de dos fases, una de distribución y otra de mezcla.
-



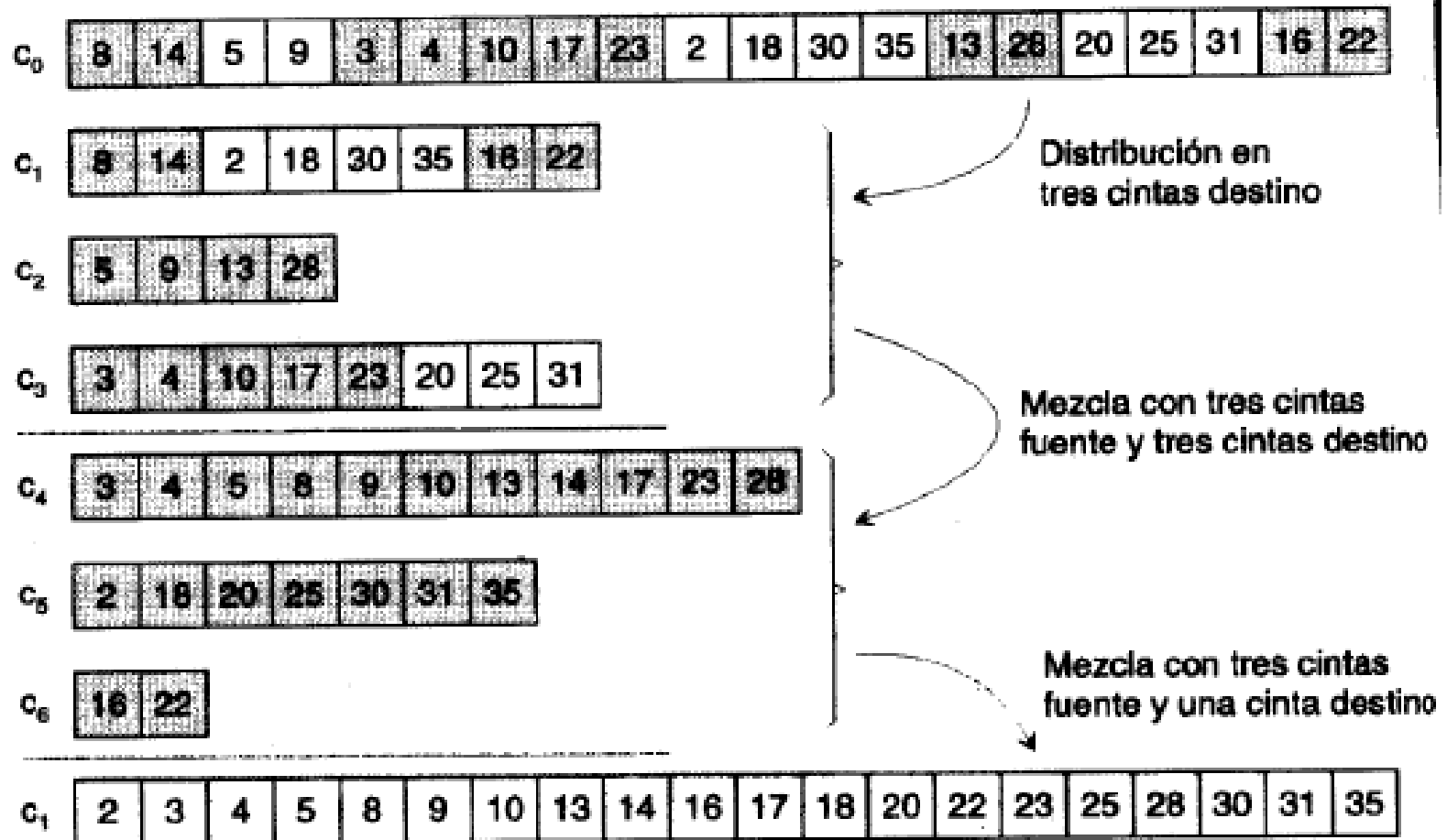
Mezcla natural. Se marcan con distinto fondo las subsecuencias ordenadas adyacentes. La barra bajo algunos elementos indica los que se han escrito durante el copiado de cabos, es decir, cuando una subsecuencia ha terminado.

Análisis mezcla natural

- En el peor caso el número de movimientos es de orden $n \log n$, e inferior en el caso promedio.
 - El número de comparaciones es mucho mayor, pero al ser el coste de una comparación muy inferior al de un movimiento, este incremento no resulta significativo.
-

3.1.3 Mezcla Balanceada Múltiple

- Persigue reducir el número de movimientos (copias) para reducir el número de pases.
 - Se consigue:
 - Realizando la distribución entre más de dos cintas (mezcla múltiple)
 - En la mezcla se combinan más de dos subsecuencias ordenadas (una por cinta)
 - La mezcla se realiza en una sola fase, sobre varias cintas auxiliares
 - Se elimina la fase de distribución, salvo la inicial
-



3.6 Mezcla balanceada con seis cintas. Se han marcado con distinto fondo las distintas subsecuencias ordenadas.

Análisis balanceada múltiple

- Hay una cinta inicial con m subsecuencias ordenadas
- Supóngase que se utilizan N cintas destino en la fase de distribución.
 - Una vez hecha la primera distribución, hay que mezclar las m subsecuencias ordenadas que están distribuidas uniformemente en N cintas
- En una primera fase de mezcla, una subsecuencia ordenada de m/N subsecuencias ordenadas, en la segunda de m/N^2 y en la k -ésima m/N^k .
- El número total de pases necesarios para clasificar n subsecuencias ordenadas con N cintas será, en el peor de los casos

$$k = \lceil \log_N n \rceil .$$

- Como cada pase necesita n copias, el número total de copias vendrá dado por

$$M = n \lceil \log_N n \rceil$$

3.1.4.- Clasificación polifásica.

- Mejora el rendimiento de la mezcla balanceada.
- Las cintas fuente y destino no son establecidas a priori, sino como consecuencia de la mezcla realizada.
 - En el proceso de mezcla, dos hacen de fuente y la tercera de destino,
 - Al finalizar las combinaciones hará de cinta destino aquella que se haya agotado, la cual sólo podrá ser identificada tras la mezcla.
- La clasificación polifásica aprovecha al máximo las cintas ya que con N cintas realiza mezclas de $N-1$ subsecuencias ordenadas.

	c_1	c_2	c_3
Distribución inicial	13	8	0
Mezcla 1	5	0	8
Mezcla 2	0	5	3
Mezcla 3	3	2	0
Mezcla 4	1	0	2
Mezcla 5	0	1	1
Mezcla 6	1	0	0

Clasificación polifásica de tres cintas.

	c_1	c_2	c_3
Distribución inicial	14	8	0
Mezcla 1	6	0	8
Mezcla 2	0	6	2
Mezcla 3	2	4	0
Mezcla 4	0	2	2
Mezcla 5	2	0	0

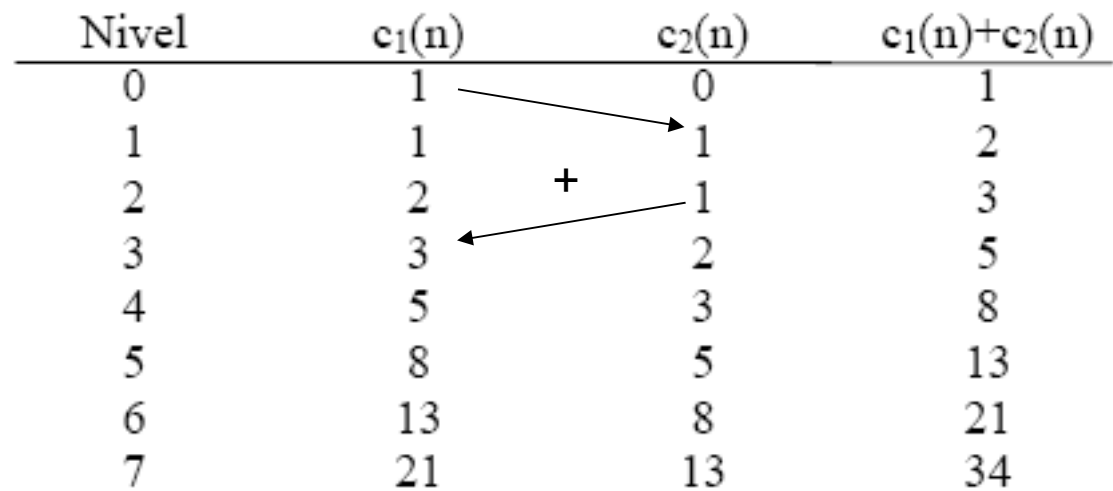
Clasificación polifásica de tres cintas no satisfactoria.

- Lo que se pretende es que al final haya una sola subsecuencia ordenada en una cinta y las demás estén agotadas. Esto no siempre es posible.
- En el ejemplo hay dos cintas vacías y a otra todavía le quedan dos subsecuencias

	C_1	C_2	C_3
Distribución inicial	14	8	0
Mezcla 1	6	0	8
Mezcla 2	0	6	2
Mezcla 3	2	4	0
Mezcla 4	0	2	2
Mezcla 5	2	0	0

Construcción de la clasificación polifásica satisfactoria de tres cintas

Nivel	$c_1(n)$	$c_2(n)$	$c_1(n)+c_2(n)$
0	1	0	1
1	1	1	2
2	2	1	3
3	3	2	5
4	5	3	8
5	8	5	13
6	13	8	21
7	21	13	34



- Para $n=0$

$$c_1(0)=1$$

$$c_2(0)=0$$

- Para cada nivel ($n>0$):

$$c_2(n+1) = c_1(n)$$

$$c_1(n+1) = c_1(n) + c_2(n) = c_1(n) + c_1(n-1)$$

- Se cumple que $c_1(n)$ son siempre números de Fibonacci y $c_2(n)$ siempre es el predecesor de $c_1(n)$.

Conclusión:

- La clasificación polifásica con 3 cintas será satisfactoria si la distribución inicial de subsecuencias ordenadas entre las cintas fuentes es tal que son **números consecutivos de Fibonacci**.
-

Nivel	$c_1(n)$	$c_2(n)$	$c_3(n)$	$c_4(n)$	$c_5(n)$	$\sum_{i=1}^5 c_i(n)$
0	1	0	0	0	0	1
1	1	1	1	1	1	5
2	2	2	2	2	1	9
3	4	4	4	3	2	17
4	8	8	7	6	4	33
5	16	15	14	12	8	65

13 Construcción de la clasificación polifásica de seis cintas satisfactoria.

Analíticamente:

$$c_5(n+1) = c_1(n)$$

$$c_4(n+1) = c_1(n) + c_5(n) = c_1(n) + c_1(n-1) \quad (3.1)$$

$$c_3(n+1) = c_1(n) + c_4(n) = c_1(n) + c_1(n-1) + c_1(n-2),$$

$$c_2(n+1) = c_1(n) + c_3(n) = c_1(n) + c_1(n-1) + c_1(n-2) + c_1(n-3)$$

$$c_1(n+1) = c_1(n) + c_2(n) = c_1(n) + c_1(n-1) + c_1(n-2) + c_1(n-3) + c_1(n-2)$$

con $c_4 = 1$,

y $c_i = 0$ para $i < 4$,

que son los números de Fibonacci de orden 4.

Condición generalizada para n cintas

- Para que una clasificación polifásica con N cintas sea satisfactoria (o perfecta) el número de subsecuencias ordenadas iniciales tiene que ser suma de cualquier $N-1, N-2, \dots, 1$ sumas de números de Fibonacci de orden $N-2$.
-

3.2.- Archivos indexados

- Las implementaciones de secuencias en algunos dispositivos de memoria secundaria, tales como discos duros, permiten un acceso cuasialeatorio (fichero):
 - Acceso aleatorio al sector + Offset de acceso secuencial
- El método de clasificación mediante archivos indexados se basa en considerar, asociado a cada llave, la dirección física del registro que caracteriza.
- Se crea un segundo archivo, archivo de índices, en el que se almacenan pares (dirección, llave).

Archivo índice

2	García
4	Núñez
5	Martín
1	Pérez
3	Ramírez
Dirección	Llave

Archivo de datos

1	Pérez			
2	García			
3	Ramírez			
4	Núñez			
5	Martín			
Dirección	Campo llave	Otros campos del fichero de datos		

Indexados

- Las operaciones de clasificación y búsqueda suelen realizarse en memoria principal sobre el archivo de índices y no sobre el archivo de datos.
- Indexado de índice denso
 - Cuando el archivo índice contiene la dirección de cada registro de la llave
- Indexado de Índice disperso
 - Cuando el archivo índice contiene grupos de llaves
 - Está formado por grupos de pares (x, b) donde b es la dirección física del bloque en el cual el primer registro tiene la llave de valor x .
 - La búsqueda se realiza por bloques lo que evita comparar con todos y cada uno de los elementos.

3.4.- Tablas de dispersión (Hashing)

- La idea básica consiste en transformar las llaves en direcciones de memoria mediante una función de transformación.
- Las tablas de dispersión se aplican cuando el conjunto de llaves posibles es mucho mayor que el de llaves reales a almacenar.
 - Los alumnos de una clase por su DNI



Definiciones I

- Dado un conjunto de llaves posibles X , y un conjunto de direcciones de memoria D , una función de transformación $H(x)$, es una aplicación suprayectiva del conjunto de llaves posibles en el conjunto de direcciones de memoria: $H: X \rightarrow D$
- El TDA tabla de dispersión es un tipo de datos homogéneo, denominadas celdas, de tamaño fijo, T_{tabla} , compuesto por un número fijo de componentes a las que se accede mediante una dirección de memoria resultante de una función de transformación. Sobre este TDA se definen los operadores Insertar, Buscar y Eliminar.
- Dos llaves distintas x_1 y x_2 son **sinónimas** si $H(x_1) = H(x_2)$

Definiciones II

■ Desbordamiento

- Cuando una nueva llave se aplica a una dirección de memoria completamente ocupada,

■ Colisión

- Cuando dos llaves distintas se aplican sobre la misma celda.

■ Densidad de llaves

- El cociente entre el número de llaves en uso, m , y el número total de llaves posibles, n_x

■ Factor de carga (o densidad de carga), α ,

- Al cociente entre el número de llaves en uso y el número total de registros almacenables en la tabla de dispersión:
 - $\alpha = m / (s*b)$; donde s es el número de registros por bloque y b el número de bloques que hay en la tabla de dispersión.

Ejemplo:

- Tabla de dispersión que consta de mil bloques con dos registros por bloque.
- El conjunto de llaves posibles es el conjunto de números del DNI (00000000 al 99999999) y
- La función de dispersión se define tal que los tres primeros dígitos del DNI asignan la dirección de memoria (000 a 999).

000	00077641	00034567
001	00131117	
549	54913110	54939189
999	99911011	99931419

3.4.1 Funciones Hash

- **Elección de una Función de Transformación**
 - Requisitos básicos de una función de transformación:
 - La dirección de memoria debe ser calculada con sencillez
 - Distribución de llaves uniforme
 - Que se produzcan el menor número posible de colisiones.
- **Función Resto de División: $f(x) = X \text{ mod } M$**
- **Función *plegado*:**
 - La dirección se obtiene dividiendo la llave en partes iguales y sumando todas ellas.
 - ***Plegado por desplazamiento:***
 - La suma de las partes puede realizarse directamente
 - ***Plegado por las fronteras :***
 - Plegar el identificador por las fronteras de las partes y sumar los dígitos coincidentes
 - Plegado en las fronteras en base decimal
 - Plegado en las fronteras en base binaria

a)

68	75	82	71	63	93	102	75
----	----	----	----	----	----	-----	----

X

	75	75	75
	102	201	102
	93	93	93
	+ 63	+ 36	+ 252
	+ 71	+ 71	+ 71
	82	28	74
	75	75	75
	68	86	34
Dirección =	629	665	776
	b)	c)	d)

e)	102 => 01100110	→	01100110 => 102
	63 => 00111111		11111100 => 252
	82 => 01010010		01001010 => 74
	68 => 01000100		00100010 => 34

- 9 a) Cadena de ocho caracteres representada por los números de orden dentro de la secuencia de cotejo correspondiente. b) Plegado por desplazamiento. c) Plegado en las fronteras en base decimal. d) Plegado en las fronteras en base binaria. e) Detalle del plegado en base binaria.

3.4.2 Manejo de desbordamiento o sobrecarga

- Definiciones
 - Colisión
 - Cuando se ha de insertar una nueva llave, si la celda que le corresponde está ocupada
 - Desbordamiento o sobrecarga
 - Si todo el bloque está lleno
 - Exploración Lineal
-

La exploración lineal

- Consiste en buscar en el bloque siguiente.
- Se distinguen dos acciones: inserción y búsqueda.
 - Para la inserción se busca en el bloque siguiente una celda libre, si vuelve a producirse sobrecarga se busca en el siguiente y así sucesivamente.
 - Para la búsqueda de una llave se busca en la dirección siguiente y así sucesivamente hasta encontrar la llave, o encontrar que la celda del bloque está vacía o la tabla está llena.
- Dirección = $f(x) + g(x)$;
 - Donde $f(x)$ es la función de dispersión y $g(x)$ es la función de tratamiento de sobrecargas.
- Considerando la tabla circular, la dirección vendrá dada por:
 - dirección = $(f(x) + g(x)) \bmod \text{TamañoTabla}$
- Las sobrecargas irán llenando los bloques cercanos a los ocupados

Exploración Cuadrática

Una técnica que mejora este comportamiento es la *exploración cuadrática*, definida por:

$$g(x) = i^2, \quad i = 1 \dots \text{TamañoTabla}$$

Con esta función de exploración la *búsqueda* se realiza examinando los bloques:

$$f(x), \quad (f(x) + i^2) \bmod \text{TamañoTabla}, \quad (f(x) - i^2) \bmod \text{TamañoTabla},$$

con $1 \leq i \leq (\text{TamañoTabla}-1) / 2$

Cuando *TamañoTabla* es un número primo de la forma $4j+3$, j entero, la exploración cuadrática puede recorrer todos los bloques de la tabla.

Rehasingh

- Técnica de generaliza los conceptos anteriores,
 - La función de exploración será una familia de funciones de dispersión que se examinan sucesivamente en un orden dado.
 - Así $g(x) = f_i(x)$, $i = 1 .. m$ donde cada $f_i(x)$ es una función de dispersión
-

Encadenamiento

- Estrategia de Encadenamiento Directo: asociar todos los elementos con dirección primaria idéntica $H(x)$ en una lista ligada. Los elementos de la lista pueden estar en la tabla primaria o no; en el segundo caso, al almacenamiento donde están asignados, se llama *área de desbordamiento*.

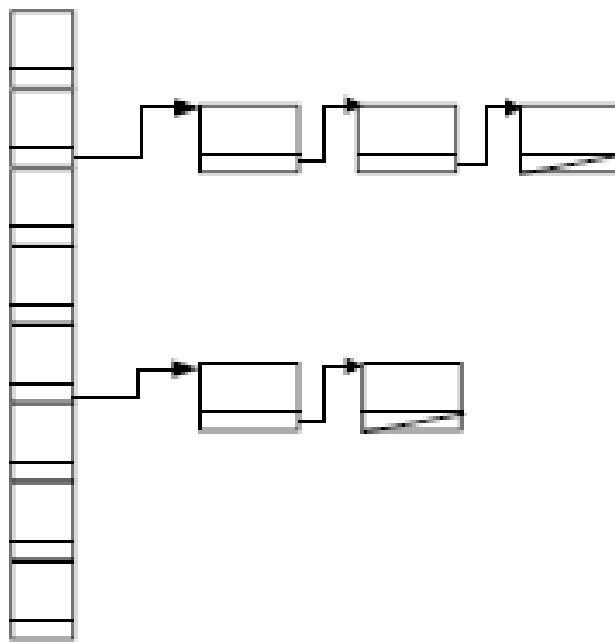


Fig.1 Con área de desbordamiento

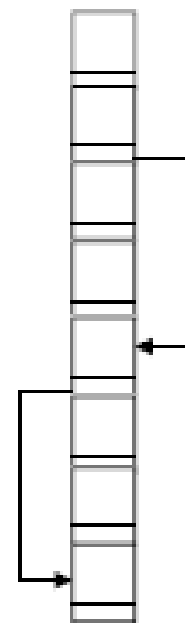


Fig.2 Sin área de desbordamiento