

# Archivos Directos

En los archivos directos, los registros se ubican mediante una función de dispersión (hashing) que aplicada a sus claves de identificación, devuelve una dirección de almacenamiento.

## 1 ORGANIZACIÓN DE REGISTROS

---

Los registros de longitud fija se pueden organizar en unidades con capacidad para uno o muchos registros

- Las de un registro se denominan **ranuras** (*slots*) o compartimientos individuales. Cuando se intenta insertar un nuevo registro en una ranura ocupada se produce una **colisión**.
- Las de múltiples registros se denominan **bloques**, cubetas (*buckets*) o compartimientos múltiples. Cuando se intenta insertar un nuevo registro en un bloque que está lleno se produce un **desborde**.

Para registros de longitud variable se deben usar bloques, para reducir la fragmentación interna.

Las ranuras deben tener un campo de control que indique si tienen contenido válido cuando se acceden.

Los bloques deben tener un campo de control que indique la cantidad de registros que contienen, si son para registros de longitud fija, o el espacio libre que les queda, si son para registros de longitud variable (en los ejemplos y ejercicios se ignora este campo para simplificarlos).

## 2 FUNCIONES DE DISPERSIÓN

---

Transforman la clave de un registro en una dirección de ranura (sólo registros de longitud fija) o en una dirección de bloque (registros de longitud fija o variable). Las direcciones son números relativos dentro del archivo.

Pueden transformar claves distintas en una misma dirección (las funciones de dispersión no son inyectivas). Las claves que resultan en una misma dirección se denominan **sinónimos**.

Para claves alfanuméricas deben **transformar los caracteres a un número**, y luego calcular el resto de dividir al número entre la cantidad total de unidades del archivo (usualmente se aplican transformaciones a los valores ASCII de todos o algunos de los caracteres).

Para claves numéricas pueden aplicar una **transformación para aleatorizar resultados** y luego calcular el resto de la división entre la cantidad total de unidades del archivo.

### 2.1 TRANSFORMACIONES DE CLAVES ALFANUMÉRICAS A NUMÉRICAS

Por ejemplo, "Info": 49|6E|66|6F (ASCII) → 73|110|102|111 (decimal)

Posibilidades (de acuerdo a la cantidad de caracteres que se tomen y al espacio de direcciones):

- Concatenar dígitos hexadecimales y pasar a decimal:  $(496E666F)_{16} \equiv 1231971951$
- Concatenar dígitos decimales: 73110102111
- Sumar valores decimales:  $73+110+102+111=396$
- Usar imaginación...

## 2.2 TRANSFORMACIONES NUMÉRICAS DE ALEATORIZACIÓN

La cantidad de dígitos a obtener debe ser de al menos dos más que la que requiera la representación de la cantidad de direcciones. En base al ejemplo anterior, se continúa con transformaciones para obtener números de seis dígitos, de los cuales se calculará el resto de la división entre la cantidad de direcciones del archivo.

### 2.2.1 Centro de Cuadrados

$1231971951^2 = 151775488050746401 \rightarrow 488805$

$73110102111^2 = 5345087030680846656321 \rightarrow 306808$

$396^2 = 156816 \rightarrow$  se toman todos los dígitos

### 2.2.2 Desplazamiento y suma

$1231971951 \rightarrow 971951 + 1231 = 973182$

$73110102111 \rightarrow 102111 + 73110 = 175221$

396 no aplica

### 2.2.3 Plegado y suma

$1231971951 \rightarrow 971951 + 132100 = 1104051$

$73110102111 \rightarrow 102111 + 11370 = 1113481$

396 no aplica

## 3 ORGANIZACIÓN DE COLISIONES Y DESBORDES

---

Los registros que colisionan en una ranura o desbordan una cubeta pueden ubicarse en otro lugar, en cuyo caso se los denomina **registros en saturación**, y van degradando paulatinamente la organización de los archivos (las búsquedas se vuelven más costosas). Las técnicas que adoptan esta solución se denominan de **dispersión estática** porque la cantidad de direcciones de dispersión de los archivos nunca cambia, y requieren la reestructuración total de los archivos cuando alcancen algún grado de saturación que se considere insostenible.

Para archivos con cubetas se pueden solucionar los desbordes mediante reestructuraciones parciales que los mantengan siempre organizados (tal como se mantienen los árboles) y que pueden implicar un cambio en la cantidad de direcciones de dispersión. Las técnicas que adoptan esta solución se denominan de **dispersión dinámica**.

### 3.1 DISPERSIÓN ESTÁTICA

#### 3.1.1 Estimación de Registros en Saturación

Archivo con **D** cubetas (espacio de direcciones  $0..D-1$ ) con capacidad para **C** registros (capacidad de cubeta), y con **R** registros almacenados en total (capacidad del archivo  $D \cdot C$ )

**Función de Poisson aplicada a la dispersión** (probabilidad que una cubeta tenga **n** registros asignados por dispersión):  $P(n) = (R/D)^n * e^{-(R/D)} / n!$

Entonces el número esperado de direcciones con **n** registros asignados es  $E(n) = D * P(n)$

El número esperado de registros en saturación (que están en cubetas que no les corresponden) es

$$S = D * (P(C+1) + 2 * P(C+2) + 3 * P(C+3) + \dots)$$

Por ejemplo, para un archivo de 1.000 cubetas con capacidad para 10 registros:

- 6.000 registros:  $S \approx 77$
- 8.000 registros:  $S \approx 420$
- 10.000 registros:  $S \approx 1.180$

A mayor carga del archivo, mayor es la cantidad de registros en saturación. El porcentaje de carga del archivo se denomina densidad de empaquetamiento  $DE = R / (D * C) \rightarrow$  para evitar alta saturación no se debe permitir DE muy alta.

También se puede comprobar que al aumentar la capacidad de las cubetas disminuye el número esperado de registros en saturación.

### 3.1.2 Organización de Registros en Saturación

Saturación en otras direcciones (direccionamiento abierto)

- Saturación lineal: búsqueda secuencial circular a partir de la ranura o cubeta siguiente; las ranuras pueden encadenarse (saturación lineal encadenada), pero las cadenas deben comenzar en la dirección que determine la función de dispersión para los sinónimos.
- Dispersión doble: una segunda función de dispersión determina la longitud de saltos para buscar ranuras o cubetas donde almacenar o encontrar registros en saturación (dispersa a los registros en saturación).

Saturación en unidades exclusivas para colisiones o desbordes (direccionamiento cerrado)

- Saturación lineal en extensión del archivo (se agregan unidades al final del archivo según se necesite).
- Encadenamiento de unidades para sinónimos de una unidad original, en otro archivo (cubetas de menor capacidad).

### 3.1.3 Caso de Estudio Comparativo

Se define la capacidad del archivo considerando que al alcanzar la máxima cantidad estimada de 20 registros su DE sería 0,8: 0,8 es a 20, lo que 1 es a la capacidad que debe tener el archivo, entonces  $\text{capacidad} = 20 / 0,8 = 25$ .

Para los archivos con ranuras se usa un campo de control para indicar su estado: vacío, ocupado o borrado. En las altas, cuando se busca un registro con clave igual al que se quiere agregar para verificar que sea único y no se esté insertando un duplicado, el estado “vacío” permite terminar búsquedas.

Si se definen, por ejemplo, bloques con capacidad para 3 registros, el archivo debe tener  $25 \div 3 = 8$  bloques, y como  $25 \bmod 3 > 0$  le sumamos uno más: 9. Se usa un campo de control en los bloques para indicar si se desbordaron o no, para permitir la finalización de búsquedas.

Se muestra un estado inicial del archivo con cubetas para tres registros, sin registros en saturación:

	Desb	R1	R2	R3
0	N	459	270	954
1	N	946	973	
2	N			

	Desb	R1	R2	R3
3	N	741	426	462
4	N			
5	N	446	410	

	Desb	R1	R2	R3
6	N	123	789	
7	N	376	484	
8	N	458	305	809

Se realizan las mismas operaciones a partir de este estado para las distintas opciones de organización de registros en saturación.

### 3.1.3.1 Saturación Lineal

#### Alta de registro con clave 630:

$630 \bmod 9 = 0$ . La cubeta 0 está llena y no desbordada: como la unicidad del registro con clave 630 está verificada, se reescribe la cubeta cambiando su estado de desborde a "S"; luego se lee la siguiente cubeta en busca de espacio y como hay, se inserta el registro y se reescribe la cubeta para persistir el cambio.

Costo de la operación: L0E0L1E1.

	Desb	R1	R2	R3
0	S	459	270	954
1	N	946	973	630
2	N			

	Desb	R1	R2	R3
3	N	741	426	462
4	N			
5	N	446	410	

	Desb	R1	R2	R3
6	N	123	789	
7	N	376	484	
8	N	458	305	809

#### Alta de registro con clave 557:

$557 \bmod 9 = 8$ . La cubeta 8 está llena y no desbordada: como la unicidad del registro con clave 557 está verificada, se reescribe la cubeta cambiando su estado de desborde a "S"; luego se leen las cubetas siguientes hasta encontrar espacio, cambiando sus estados de desborde si fuera necesario.

Costo de la operación: L8E8L0L1E1L2E2.

	Desb	R1	R2	R3
0	S	459	270	954
1	S	946	973	630
2	N	557		

	Desb	R1	R2	R3
3	N	741	426	462
4	N			
5	N	446	410	

	Desb	R1	R2	R3
6	N	123	789	
7	N	376	484	
8	S	458	305	809

### Alta de registro con clave 731:

$731 \bmod 9 = 2$ . Como la cubeta no está desbordada y no contiene ningún registro con clave 731, se puede asegurar que no hay otro registro en el archivo con clave 731.

Costo de la operación: L2E2.

	Desb	R1	R2	R3
0	S	459	270	954
1	S	946	973	630
2	N	557	731	

	Desb	R1	R2	R3
3	N	741	426	462
4	N			
5	N	446	410	

	Desb	R1	R2	R3
6	N	123	789	
7	N	376	484	
8	S	458	305	809

### Búsqueda de registro con clave 557 (en estado previo):

$557 \bmod 9 = 8$ . La cubeta 8 no contiene a ningún registro con esa clave pero está desbordada: se busca a partir de la siguiente (circular), hasta encontrarlo o alcanzar una cubeta NO desbordada que NO lo contenga.

Costo: L8L0L1L2.

### Baja de registro con clave 459:

$459 \bmod 9 = 0$ .

Costo de la operación: L0E0.

Observaciones:

No se cambia el indicador de desborde de la cubeta, porque impediría encontrar registros en saturación que le correspondan. Se podría buscar linealmente un registro en saturación correspondiente a la cubeta para ponerlo en su dirección base y optimizar búsquedas.

	Desb	R1	R2	R3
0	S	270	954	
1	S	946	973	630
2	N	557	731	

	Desb	R1	R2	R3
3	N	741	426	462
4	N			
5	N	446	410	

	Desb	R1	R2	R3
6	N	123	789	
7	N	376	484	
8	S	458	305	809

#### 3.1.3.2 Saturación Lineal Encadenada (sólo archivos con ranuras)

Se considera un estado inicial para el archivo con los mismos registros que el archivo de cubetas. El campo Sig contiene el número de ranura con el siguiente sinónimo.

	Estado	Contenido	Sig
0	V		
1	O	376	2
2	O	426	
3	V		
4	O	954	
5	O	305	
6	V		
7	V		
8	O	458	
9	O	459	11

	Estado	Contenido	Sig
10	O	410	
11	O	484	13
12	O	462	
13	O	809	
14	O	789	
15	V		
16	O	741	
17	V		
18	V		
19	V		

	Estado	Contenido	Sig
20	O	270	
21	O	946	22
22	O	446	
23	O	123	24
24	O	973	

#### Alta de registro con clave 630:

$630 \bmod 25 = 5$ . La ranura 5 está ocupada por un registro cuya clave es sinónimo de 630, puesto que  $305 \bmod 25 = 5$ ; entonces se busca linealmente a partir de la ranura 5 (ya que no tiene ningún sinónimo siguiente y la unidad del registro que se quiere insertar está asegurada) a otra que esté vacía o borrada, y se pondrá el número de la encontrada como dirección del siguiente sinónimo en la 5.

Costo: L5L6E6E5.

	Estado	Contenido	Sig
0	V		
1	O	376	2
2	O	426	
3	V		
4	O	954	
5	O	305	6
6	O	630	
7	V		
8	O	458	
9	O	459	11

	Estado	Contenido	Sig
10	O	410	
11	O	484	13
12	O	462	
13	O	809	
14	O	789	
15	V		
16	O	741	
17	V		
18	V		
19	V		

	Estado	Contenido	Sig
20	O	270	
21	O	946	22
22	O	446	
23	O	123	24
24	O	973	

#### Alta de registro con clave 557:

$557 \bmod 25 = 7$ . La ranura 7 está vacía, por lo que se almacena el registro en ella, sabiendo que no hay otro en el archivo con la misma clave.

Costo: L7E7.

	Estado	Contenido	Sig
0	V		
1	O	376	2
2	O	426	
3	V		
4	O	954	
5	O	305	6
6	O	630	

	Estado	Contenido	Sig
7	O	557	
8	O	458	
9	O	459	11
10	O	410	
11	O	484	13
12	O	462	
13	O	809	

	Estado	Contenido	Sig
14	O	789	
15	V		
16	O	741	
17	V		
18	V		
19	V		
20	O	270	

	<b>Estado</b>	<b>Contenido</b>	<b>Sig</b>
21	O	946	22
22	O	446	

	<b>Estado</b>	<b>Contenido</b>	<b>Sig</b>
23	O	123	24
24	O	973	

#### Alta de registro con clave 731:

$731 \bmod 25 = 6$ . La ranura 6 está ocupada por un registro cuya clave no es sinónimo de 731, ya que  $630 \bmod 25 = 5$  (el registro con clave es un intruso); entonces se busca una ranura vacía o borrada a partir de la 6 para cambiar de lugar al registro intruso; luego se buscará la ranura que tenga como siguiente al intruso recorriendo su propia cadena de sinónimos para actualizarla y se almacena el registro con clave 731 en la ranura 6, sabiendo que no hay otro registro en el archivo con la misma clave.

Costo: L6L7L8L9L10L11L12L13L14L15E15L5E5E6.

	<b>Estado</b>	<b>Contenido</b>	<b>Sig</b>
0	V		
1	O	376	2
2	O	426	
3	V		
4	O	954	
5	O	305	15
6	O	731	
7	O	557	
8	O	458	
9	O	459	11

	<b>Estado</b>	<b>Contenido</b>	<b>Sig</b>
10	O	410	
11	O	484	13
12	O	462	
13	O	809	
14	O	789	
15	O	630	
16	O	741	
17	V		
18	V		
19	V		

	<b>Estado</b>	<b>Contenido</b>	<b>Sig</b>
20	O	270	
21	O	946	22
22	O	446	
23	O	123	24
24	O	973	

#### Búsqueda de registro con clave 557 (en estado previo):

$557 \bmod 25 = 7$ .

Costo: L7.

#### Baja de registro con clave 459:

$459 \bmod 25 = 9$ . Como se encuentra al registro en la ranura 9 y ésta tiene la dirección de una siguiente, se trae al siguiente registro de la cadena para que la dirección inicial esté ocupada.

Costo: L9L11E9E11.

	<b>Estado</b>	<b>Contenido</b>	<b>Sig</b>
0	V		
1	O	376	2
2	O	426	
3	V		
4	O	954	
5	O	305	15
6	O	731	
7	O	557	
8	O	458	

	<b>Estado</b>	<b>Contenido</b>	<b>Sig</b>
9	O	484	13
10	O	410	
11	B	484	13
12	O	462	
13	O	809	
14	O	789	
15	O	630	
16	O	741	
17	V		

	<b>Estado</b>	<b>Contenido</b>	<b>Sig</b>
18	V		
19	V		
20	O	270	
21	O	946	22
22	O	446	
23	O	123	24
24	O	973	

### 3.1.3.3 Dispersión Doble

Segunda función  $D(k)=k \bmod 5 + 1$  para saltos de búsqueda.

Algorítmicamente se puede integrar a las dos funciones en la fórmula  $h(k) = (k \bmod 9 + i * (k \bmod 5 + 1)) \bmod 9$ , con  $i = 0, 1, 2, \dots$  (i implica el número de intentos de búsqueda de espacio libre para registros en saturación).

#### Alta de registro con clave 630:

$(630 \bmod 9 + 0 * (630 \bmod 5 + 1)) \bmod 9 = 0$ . Como la cubeta 0 está llena, se cambia su estado a desbordada y se calcula la función para  $i = 1$  en busca de otra con espacio libre:  $(630 \bmod 9 + 1 * (630 \bmod 5 + 1)) \bmod 9 = 1$ ; como la cubeta 1 tiene lugar, se inserta el registro en ella.

Costo: L0E0L1E1.

	Desb	R1	R2	R3
0	S	459	270	954
1	N	946	973	630
2	N			

	Desb	R1	R2	R3
3	N	741	426	462
4	N			
5	N	446	410	

	Desb	R1	R2	R3
6	N	123	789	
7	N	376	484	
8	N	458	305	809

#### Alta de registro con clave 557:

$(557 \bmod 9 + 0 * (557 \bmod 5 + 1)) \bmod 9 = 8$ . La cubeta 8 está llena y no desbordada: se reescribe cambiando su estado a desbordada y se recalcula la función para  $i = 1$ :  $(557 \bmod 9 + 1 * (557 \bmod 5 + 1)) \bmod 9 = (8 + 3) \bmod 9 = 2$ ; como la cubeta 2 tiene lugar, se inserta el registro en ella.

Costo: L8E8L2E2 (comparar con saturación lineal).

	Desb	R1	R2	R3
0	S	459	270	954
1	N	946	973	630
2	N	557		

	Desb	R1	R2	R3
3	N	741	426	462
4	N			
5	N	446	410	

	Desb	R1	R2	R3
6	N	123	789	
7	N	376	484	
8	S	458	305	809

#### Alta de registro con clave 731:

$(731 \bmod 9 + 0 * (731 \bmod 5 + 1)) \bmod 9 = 2$ .

Costo: L2E2.



	Desb	R1	R2	R3
0	S	459	270	954
1	N	946	973	630
2	N	557	731	

	Desb	R1	R2	R3
3	N	741	426	462
4	N			
5	N	446	410	

	Desb	R1	R2	R3
6	N	123	789	
7	N	376	484	
8	S	458	305	809

#### Búsqueda de registro con clave 557 (en estado previo):

$(557 \bmod 9 + 0 * (557 \bmod 5 + 1)) \bmod 9 = 8$ . La cubeta 8 no contiene a ningún registro con esa clave pero está desbordada; se recalcula la función aumentando i, hasta encontrarlo o alcanzar una cubeta NO desbordada que NO lo contenga:  $(557 \bmod 9 + 1 * (557 \bmod 5 + 1)) \bmod 9 = 2$ .

Costo: L8L2.

#### Baja de registro con clave 459:

$(459 \bmod 9 + 0 * (459 \bmod 5 + 1)) \bmod 9 = 0$ .

No se cambia el indicador de desborde de la cubeta, porque impediría encontrar registros en saturación que le correspondan. Se podría buscar **linealmente**, dado que los registros en saturación están dispersos, un registro en saturación correspondiente a la cubeta para ponerlo en su dirección base y optimizar búsquedas, pero en este caso, una cubeta no desbordada no implica que no se vaya a encontrar más adelante un desborde...

Costo: LOE0.

	Desb	R1	R2	R3
0	S	270	954	
1	N	946	973	630
2	N	557	731	

	Desb	R1	R2	R3
3	N	741	426	462
4	N			
5	N	446	410	

	Desb	R1	R2	R3
6	N	123	789	
7	N	376	484	
8	S	458	305	809

#### 3.1.3.4 Saturación Lineal en Extensión del Archivo

##### Alta de registro con clave 630 (partiendo de estado inicial sin saturación):

$630 \bmod 9 = 0$ . La cubeta 0 está llena y no desbordada: se reescribe cambiando su estado de desborde a "S". Se agrega una cubeta al final del archivo. **Se ignora el campo de desborde en el área de saturación.**

Costo de la operación: LOE0E9

	Desb	R1	R2	R3
0	S	459	270	954
1	N	946	973	

	Desb	R1	R2	R3
2	N			
3	N	741	426	462

	Desb	R1	R2	R3
4	N			
5	N	446	410	

	Desb	R1	R2	R3
6	N	123	789	
7	N	376	484	

	Desb	R1	R2	R3
8	N	458	305	809
9		630		

#### Alta de registro con clave 557:

$557 \bmod 9 = 8$ . La cubeta 8 está llena y no desbordada: se reescribe cambiando su estado de desborde a "S".  
Se lee la primer cubeta de desborde: 9.

Costo de la operación: L8E8L9E9.

	Desb	R1	R2	R3
0	S	459	270	954
1	N	946	973	
2	N			
3	N	741	426	462

	Desb	R1	R2	R3
4	N			
5	N	446	410	
6	N	123	789	
7	N	376	484	

	Desb	R1	R2	R3
8	S	458	305	809
9		630	557	

#### Alta de registro con clave 731:

$731 \bmod 9 = 2$ .

Costo de la operación: L2E2.

	Desb	R1	R2	R3
0	S	459	270	954
1	N	946	973	
2	N	731		
3	N	741	426	462

	Desb	R1	R2	R3
4	N			
5	N	446	410	
6	N	123	789	
7	N	376	484	

	Desb	R1	R2	R3
8	S	458	305	809
9		630	557	

#### Búsqueda de 557 (en estado previo):

$557 \bmod 9 = 8$ . La cubeta 8 no contiene a ningún registro con esa clave pero está desbordada: se busca a partir de la primer cubeta de desborde (9), hasta encontrarlo o alcanzar la última cubeta del archivo.

Costo: L8L9.

#### Baja de registro con clave 459:

$459 \bmod 9 = 0$ . No se cambia el indicador de desborde de la cubeta, porque impediría encontrar registros en saturación que le correspondan. Se podría buscar linealmente en el área de desborde un registro en saturación correspondiente a la cubeta para ponerlo en su dirección base y optimizar búsquedas.

Costo de la operación: LOE0.

	Desb	R1	R2	R3
0	S	270	954	
1	N	946	973	
2	N	731		
3	N	741	426	462

	Desb	R1	R2	R3
4	N			
5	N	446	410	
6	N	123	789	
7	N	376	484	

	Desb	R1	R2	R3
8	S	458	305	809
9		630	557	

### 3.1.3.5 Encadenamiento de Cubetas para Sinónimos de una Cubeta Original en otro Archivo

En esta variante, el indicador de desborde se sustituye por un número relativo de cubeta del archivo de desbordes (-1 indica que no hay desbordes).

Se considera un archivo de desbordes con ranuras, y en caso de que haya más de un desborde de una misma cubeta, las ranuras que ocupen se encadenan. Las ranuras requieren un estado (Ocupado o Libre) para control de espacio libre. Si en lugar de ranuras fueran cubetas, con el contador de registros alcanza.

Alta 630              LOE0(datos)-E0(desbordes)

Alta 557              L8E8(datos)-E1(desbordes)

Alta 731              L2E2(datos)

Búsqueda 557      L8(datos)-L1(desbordes)

### Archivo de Datos

	Desb	R1	R2	R3
0	0	459	270	954
1	-1	946	973	
2	-1			

	Desb	R1	R2	R3
3	-1	741	426	462
4	-1			
5	-1	446	410	

	Desb	R1	R2	R3
6	-1	123	789	
7	-1	376	484	
8	1	458	305	809

### Archivo de Desbordes

	Estado	Sig	R
0	O	-1	630
1	O	-1	557

Baja 459      LOE0(datos)

Se podría traer un desborde de la cubeta 0 para optimizar búsquedas en el archivo de datos.

#### Archivo de Datos

	Desb	R1	R2	R3
0	0	270	954	
1	-1	946	973	
2	-1			

	Desb	R1	R2	R3
3	-1	741	426	462
4	-1			
5	-1	446	410	

	Desb	R1	R2	R3
6	-1	123	789	
7	-1	376	484	
8	1	458	305	809

#### Archivo de Desbordes

	Estado	Sig	R
0	O	-1	630
1	O	-1	557

### 3.2 DISPERSIÓN DINÁMICA

Existen varias técnicas de dispersión dinámica, siendo las más recomendadas las de Dispersión Extensible:

- La función de dispersión se usa para acceder a una tabla (archivo) de direcciones de cubetas, o de dispersión
- Cada vez que se desborda una cubeta del archivo, se agrega una nueva y se distribuye la carga de la desbordada entre ésta y la nueva mediante la función de dispersión
- De tanto en tanto, cuando se agrega una cubeta al archivo la tabla de dispersión se duplica
- Las cubetas requieren campos de control para indicar la cantidad de registros (de longitud fija) o la cantidad de bytes libres (registros de longitud variable), y para indicar la cantidad de bits invariables de la clave de identificación de los registros que se considera para acceder a la tabla de dispersión (bits de dispersión)

La versión de Dispersión Extensible que veremos en esta asignatura se denomina **Modular** o **De Bits Sufijos** porque se considera a los últimos bits de la clave de identificación de un registro para determinar el número de registro de la tabla de dispersión que contiene el número de cubeta del archivo de datos donde ubicar al registro.

El tamaño de la tabla de dispersión determina la cantidad de bits a considerar en las claves para acceder a la tabla: si el **tamaño de la tabla** en cantidad de registros es **T**, la cantidad de bits a considerar para la dispersión es **Log<sub>2</sub>(T)**, lo cual es equivalente a calcular el **módulo T** de las claves.

Para la resolución de desbordes se requiere conocer, de la cubeta que se desborda, la cantidad de bits invariables de la clave de identificación de sus registros que se considera para acceder a la tabla de dispersión (sus bits de dispersión).

Para comparar resultados con los de las variantes de dispersión estática, se considera la inserción de los mismos registros del estado inicial del caso de estudio para dispersión estática, en el siguiente orden: 270, 946, 741, 446, 123, 376, 458, 954, 973, 426, 410, 789, 484, 305, 462, 809, 459. El archivo de datos es de cubetas con capacidad para tres registros.

Para insertar los tres primeros registros no se considera ningún bit de las claves de identificación de los registros, lo que equivale a calcularles el módulo 1 (el tamaño de la tabla de dispersión es de un único registro, y  $\log_2(1)=0$ ):

Tabla:

0
0

Archivo:

	Bits	R1	R2	R3
0	0	270	946	741

Al aparecer el cuarto registro, con clave **446** ( $446 \bmod 1 = 0$ , el registro 0 de la tabla refiere a la cubeta 0), se produce el desborde de la cubeta 0: si los bits de dispersión de la cubeta desbordada coinciden con  $\log_2(T)$ , T tamaño de la tabla, se duplica la tabla y en el registro de la tabla que refiere a la cubeta desbordada antes de la duplicación, se escribe el número de una cubeta nueva del archivo de datos; luego se dispersan los registros de la cubeta desbordada para la nueva tabla, y finalmente se insiste con el alta del registro que produjo el desborde luego de la reorganización parcial; se suma uno a los bits de dispersión de la cubeta desbordada, y se copia este nuevo valor para el bloque nuevo.

Tabla:

1	0
0	1

Archivo:

	Bits	R1	R2	R3
0	1	741		
1	1	270	946	446

$270 \bmod 2 = 0$  (cubeta 1);  $946 \bmod 2 = 0$  (cubeta 1);  $741 \bmod 2 = 1$  (cubeta 0); **446**  $\bmod 2 = 0$  (cubeta 1).

El costo de la operación es L0E0E1.

Observar que el cálculo del módulo 2 de una clave resulta en un número que se representa con  $\log_2(2)=1$  bit (el último) de la clave:

$$270 \equiv (100001110)_2; 270 \bmod 2 = 0; 0 \equiv (0)_2$$

$$741 \equiv (1011100101)_2; 741 \bmod 2 = 1; 1 \equiv (1)_2$$

**Alta del registro con clave 123:**

$123 \bmod 2 = 1$ ; el registro 1 de la tabla refiere a la cubeta 0. Costo: L0E0.

Tabla:

1	0
0	1

Archivo:

	Bits	R1	R2	R3
0	1	741	123	
1	1	270	946	446

### Alta del registro con clave 376:

$376 \bmod 2 = 0$ ; el registro 0 de la tabla refiere a la cubeta 1, que se desborda. Se efectúa una reorganización parcial de la cubeta 1, que al tener tantos bits de dispersión como los necesarios para direccionar registros de la tabla ( $\log_2(2)=1$ ), determina la duplicación de la tabla.

Tabla:

2	0	1	0
0	1	2	3

Archivo:

	Bits	R1	R2	R3
0	1	741	123	
1	2	270	946	446
2	2	376		

$270 \bmod 4 = 2$  (queda en cubeta 1);  $946 \bmod 4 = 2$  (queda en cubeta 1);  $446 \bmod 4 = 2$  (queda en cubeta 1);  $376 \bmod 4 = 0$  (va a cubeta 2).

Costo: L1E1E2.

### Alta del registro con clave 458:

$458 \bmod 4 = 2$ ; el registro debe almacenarse en la cubeta 1, pero como está llena se debe hacer una reorganización parcial. Como la cantidad de bits de dispersión de la cubeta 1 es la misma que se necesita para direccionar registros de la tabla, la tabla se duplica, y en el registro 2 de la tabla se refiere a una nueva cubeta. Se incrementa en uno la cantidad de bits de dispersión de la cubeta 1, y a la nueva cubeta 3 se le pone esta última cantidad de bits de dispersión.

Tabla:

2	0	3	0	2	0	1	0
0	1	2	3	4	5	6	7

Archivo:

	Bits	R1	R2	R3
0	1	741	123	
1	3	270	446	
2	2	376		
3	3	946	458	

$270 \bmod 8 = 6$  (cubeta 1);  $946 \bmod 8 = 2$  (cubeta 3);  $446 \bmod 8 = 6$  (cubeta 1);  $458 \bmod 8 = 2$  (cubeta 3).

Costo: L1E1E3.

### Alta del registro con clave 954:

$954 \bmod 8 = 2$ ; el registro 2 de la tabla refiere a la cubeta 3. Costo: L3E3.

Tabla:

2	0	3	0	2	0	1	0
0	1	2	3	4	5	6	7

Archivo:

	Bits	R1	R2	R3
0	1	741	123	
1	3	270	446	
2	2	376		

3	3	946	458	954
---	---	-----	-----	-----

**Alta del registro con clave 973:**

$973 \bmod 8 = 5$ ; cubeta 0. Costo L0E0.

Tabla:

2	0	3	0	2	0	1	0
0	1	2	3	4	5	6	7

Archivo:

	Bits	R1	R2	R3
0	1	741	123	973
1	3	270	446	
2	2	376		
3	3	946	458	954

**Alta del registro con clave 426:**

$426 \bmod 8 = 2$ ; cubeta 3 que se desborda.

Tabla:

2	0	4	0	2	0	1	0	2	0	3	0	2	0	1	0
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Archivo:

	Bits	R1	R2	R3
0	1	741	123	973
1	3	270	446	
2	2	376		
3	4	458	954	426
4	4	946		

$946 \bmod 16 = 2$  (pasa a cubeta 4);  $458 \bmod 16 = 10$  (queda en cubeta 3);  $954 \bmod 16 = 10$  (queda en cubeta 3); **426**  $\bmod 16 = 10$  (va a cubeta 3). Costo: L3E3E4.

**Alta del registro con clave 410:**

$410 \bmod 16 = 10$ ; cubeta 3 que vuelve a desbordarse.

Tabla:

2	0	4	0	2	0	1	0	2	0	5	0	2	0	1	0
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

2	0	4	0	2	0	1	0	2	0	3	0	2	0	1	0
16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31

Archivo:

Bits	R1	R2	R3
------	----	----	----

0	1	741	123	973
1	3	270	446	
2	2	376		
3	5	954	410	
4	4	946		
5	5	458	426	

$458 \bmod 32 = 10$  (pasa a cubeta 5);  $954 \bmod 32 = 26$  (queda en cubeta 3);  $426 \bmod 32 = 10$  (pasa a cubeta 5); **410**  $\bmod 32 = 26$  (va a cubeta 3). Costo: L3E3E5.

#### Alta del registro con clave 789:

$789 \bmod 32 = 21$ ; cubeta 0 que se desborda y tiene menos bits de dispersión que los necesarios para acceder a la tabla: no es necesario duplicar la tabla, porque la cubeta 0 está referida varias veces en ella (si  $T$  es el tamaño de la tabla y  $B$  los bits de dispersión de una cubeta, entonces el número de esa cubeta está  $T/2^B$  veces en la tabla, siempre a  $2^B$  registros de distancia). Se aumenta en 1 los bits de dispersión de la cubeta 0 que quedan en **2**, y se agrega otra cubeta al archivo con esta nueva cantidad de bits de dispersión; a partir de la posición 21, y en un recorrido circular de la tabla a saltos de tamaño  $2^2 = 4$  registros, se cambia la referencia a la cubeta 0 por una referencia a la cubeta nueva 6 (se deben cambiar  $32/4 = 8$  referencias). Luego se redispersan los registros de la cubeta 0 y se vuelve a intentar el alta. Costo: L0E0E6.

Tabla:

2	<b>6</b>	4	0	2	<b>6</b>	1	0	2	<b>6</b>	5	0	2	<b>6</b>	1	0
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

2	<b>6</b>	4	0	2	<b>6</b>	1	0	2	<b>6</b>	3	0	2	<b>6</b>	1	0
16	17	18	19	20	<b>21</b>	22	23	24	25	26	27	28	29	30	31

Archivo:

	Bits	R1	R2	R3
0	<b>2</b>	123		
1	3	270	446	
2	2	376		
3	5	954	410	
4	4	946		
5	5	458	426	
6	<b>2</b>	741	973	<b>789</b>

$741 \bmod 32 = 5$  (pasa a cubeta 6);  $123 \bmod 32 = 27$  (queda en cubeta 0);  $973 \bmod 32 = 13$  (pasa a cubeta 6); para el registro nuevo ya se calculó que va a la cubeta referida en la posición 21, que ahora es la 6.

Costo: L0E0E6.



### Observación:

$789 \equiv (1100010101)_2$ ; como  $\log_2(32)=5$ , la función de dispersión consiste en tomar los últimos 5 bits de las claves:  ~~$(1100010101)_2 \equiv 21$~~ , pero como la cubeta referida en el registro 21 de la tabla tiene 1 solo bit de dispersión, significa que las claves de registros en esa cubeta tienen invariante al último bit, y que los registros de la tabla que refieren a esa cubeta están en las posiciones determinadas por todas las variaciones de los otros cuatro bits: 0000**1**, 0001**1**, 0010**1**, 0011**1**, ... 1111**1** (1, 3, 5, 7, ... 31). Como al desbordarse la cubeta se debe agregar otra, las referencias es esta última en la tabla tienen que quedar en las posiciones determinadas al hacer invariante a un bits más de las claves, en esta caso a los últimos dos: 0000**11**, 0010**11**, 0100**11**, 0110**11**, 1000**11**, 1010**11**, 1100**11**, 1110**11** (1, 5, 9, 13, 17, 21, 25, 29).

### Alta del registro con clave 484:

$484 \bmod 32 = 4$ ; cubeta 2. Costo L2E2.

Tabla:

2	6	4	0	2	6	1	0	2	6	5	0	2	6	1	0
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

  

2	6	4	0	2	6	1	0	2	6	3	0	2	6	1	0
16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31

Archivo:

	Bits	R1	R2	R3
0	2	123		
1	3	270	446	
2	2	376	484	
3	5	954	410	
4	4	946		
5	5	458	426	
6	2	741	973	789

### Alta del registro con clave 305:

$305 \bmod 32 = 17$ ; cubeta 6 que se desborda y que tiene menos bits de dispersión que los de la tabla ( $\log_2(32)=5$ ). Se aumenta en 1 los bits de dispersión de la cubeta 6 que quedan en **3**, se agrega una cubeta nueva 7 con 3 bits de dispersión, y a partir de la posición 17 de la tabla se cambia el número de cubeta 6 por el número 7, a saltos de  $2^3 = 8$  registros (en total  $32/8 = 4$  cambios); luego se redispersa el contenido de la cubeta 6 y se reintenta el alta.

Costo: L6E6E7.

Tabla:

2	7	4	0	2	6	1	0	2	7	5	0	2	6	1	0
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

2	7	4	0	2	6	1	0	2	7	3	0	2	6	1	0
16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31

Archivo:

	Bits	R1	R2	R3
0	2	123		
1	3	270	446	
2	2	376	484	
3	5	954	410	
4	4	946		
5	5	458	426	
6	3	741	973	789
7	3	305		

$741 \bmod 32 = 5$  (queda en cubeta 6);  $973 \bmod 32 = 13$  (queda en cubeta 6);  $789 \bmod 32 = 21$  (queda en cubeta 6); y el nuevo registro va a la cubeta 7.

**Alta de registros con claves 462, 809 y 459:**

$462 \bmod 32 = 14$ ; cubeta 1. Costo: L1E1.

$809 \bmod 32 = 9$ ; cubeta 7. Costo: L7E7.

$459 \bmod 32 = 11$ ; cubeta 0. Costo: L0E0.

Tabla:

2	7	4	0	2	6	1	0	2	7	5	0	2	6	1	0
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

2	7	4	0	2	6	1	0	2	7	3	0	2	6	1	0
16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31

Archivo:

	Bits	R1	R2	R3
0	2	123	459	
1	3	270	446	462
2	2	376	484	
3	5	954	410	
4	4	946		
5	5	458	426	
6	3	741	973	789
7	3	305	809	

Como las operaciones del caso de estudio no aportan nueva información de esta técnica, se considera una secuencia de bajas para mostrar distintos escenarios problemáticos de esta primitiva.

En las altas, los escenarios problemáticos son los desbordes, que pueden ocurrir con cubetas que tengan igual cantidad de bits de dispersión que los necesarios para representar direcciones de registros de la tabla de dispersión ( $\log_2(t)$  bits, con T tamaño de la tabla), o con cubetas que tengan menor cantidad de bits; la diferencia para resolver el alta en uno u otro escenario es en la actualización de la tabla. Pero hay una situación que no ocurrió en las altas realizadas previamente: puede suceder que la dispersión de registros de una cubeta que se desborda resulte en que todos los registros queden en la cubeta que se desborda o bien que todos pasen a la nueva, y que al reintentar el alta luego de la redispersión o reorganización parcial, se produzca otro desborde. En este caso, se debe proceder en forma recursiva: se agrega otra cubeta y se efectúa la reorganización parcial para solucionar desbordes tal como se vio, tantas veces cuantas sea necesario para insertar el nuevo registro. Cada vez que una reorganización parcial resulta inútil porque tras la misma vuelve a haber un desborde, se observará que en el archivo queda una cubeta totalmente vacía referida en la tabla.

En las bajas, los escenarios problemáticos o de excepción aparecen cuando tras una de estas operaciones una cubeta queda totalmente vacía: puede suceder que esa cubeta se pueda liberar sin problemas sustituyendo toda referencia de ella en la tabla por referencias a otra cubeta, o puede suceder que sea imposible liberarla por la configuración de la tabla (no se puede sustituir referencias sin que la tabla quede inconsistente). Siempre que se libere una cubeta con la misma cantidad de bits de dispersión que los bits para direccionar la tabla, debe validarse que tras la operación, la tabla no haya quedado duplicada (dos mitades iguales), y si esto sucediera se debe acortarla desechando la segunda mitad.

#### Baja de registros con claves 305 y 809:

$305 \bmod 32 = 17$ ; cubeta 7. Costo: L7E7 (baja ordinaria).

$809 \bmod 32 = 9$ ; cubeta 7. Como la cubeta 7 queda vacía, debe verificarse si es posible liberarla: se consideran sus bits de dispersión menos 1 ( $3-1=2$ ); se eleva a 2 a esa cantidad de bits para obtener una distancia de verificación de la tabla ( $2^2=4$ ), y se leen los registros a esa distancia circular antes y después de la posición 9 (los registros  $(9-4) \bmod 32=5$  y  $(9+4) \bmod 32=13$ ) y si ambos refieren a una misma cubeta entonces es posible liberarla (como en este caso); en caso contrario no se puede liberar. La liberación de la cubeta implica leer la cubeta de reemplazo en la tabla (la 6), disminuir en uno sus bits de dispersión y reescribirla con este único cambio, y sustituir las referencias a la cubeta que se libera por referencias a la de reemplazo ( $32/2^3$  reemplazos a distancia  $2^3$  a partir del registro 9). El bloque liberado no se reescribe.

Costo: L7L6E6.

Tabla:

2	6	4	0	2	6	1	0	2	6	5	0	2	6	1	0
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

2	6	4	0	2	6	1	0	2	6	3	0	2	6	1	0
16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31

Archivo:

	Bits	R1	R2	R3
0	2	123	459	
1	3	270	446	462
2	2	376	484	
3	5	954	410	
4	4	946		
5	5	458	426	
6	2	741	973	789
7	3	809		

**Cubetas Libres: 7**

**Baja de registro con clave 946:**

$946 \bmod 32 = 18$ ; cubeta 4. Como sus bits de dispersión son 4, se verifican los números de cubeta en las posiciones  $18-2^{4-1}=10$  y  $18+2^{4-1}=26$  de la tabla. Como no coinciden, la cubeta 4 no puede liberarse: se debe reescribir vacía.

Costo: L4E4.

Tabla:

2	6	4	0	2	6	1	0	2	6	5	0	2	6	1	0
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

2	6	4	0	2	6	1	0	2	6	3	0	2	6	1	0
16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31

Archivo:

	Bits	R1	R2	R3
0	2	123	459	
1	3	270	446	462
2	2	376	484	
3	5	954	410	
4	4			
5	5	458	426	
6	2	741	973	789
7	3	809		

**Cubetas Libres: 7**

**Baja de registros con claves 410 y 954:**

$410 \bmod 32 = 26$ ; cubeta 3. Costo: L3E3 (baja ordinaria).

$954 \bmod 32 = 26$ ; cubeta 3. Como sus bits de dispersión son 5, se verifican los números de cubeta en las posiciones  $26 - 2^{5-1} = 10$  y  $(26 + 2^{5-1}) \bmod 32 = 10$  de la tabla. Como coinciden, se puede liberar la cubeta, y como los bits de dispersión de la cubeta son iguales a los bits para direccionar la tabla, se verifica si la primera mitad de la tabla es igual a la segunda: como quedan iguales, se trunca la tabla a la mitad.

Costo: L3L5E5.

Tabla:

2	6	4	0	2	6	1	0	2	6	5	0	2	6	1	0
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

(tabla truncada en la posición 16)

2	6	4	0	2	6	1	0	2	6	5	0	2	6	1	0
16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31

Archivo:

	Bits	R1	R2	R3
0	2	123	459	
1	3	270	446	462
2	2	376	484	
3	5	954		
4	4			
5	4	458	426	
6	2	741	973	789
7	3	809		

**Cubetas Libres: 7, 3**

### 3.3 EJERCICIOS

- Sea un **archivo directo** con cubetas para dos registros y organización de desbordes por saturación lineal.
  - Cárguelo al 70% con las claves 1239, 2461, 1837, 2246, 2149, 991 y 2547, y muestre su estado (determinar el tamaño del archivo considerando que con 7 registros su densidad de empaquetamiento debe ser del 70%).
  - Elimine la clave 2461, muestre el estado del archivo tras la operación e indique qué cubetas leyó y escribió para efectuarla.
  - Inserte la clave 1001, muestre el estado del archivo tras la operación e indique qué cubetas leyó y escribió para efectuarla.
- Dado un archivo directo con cubetas para dos registros y saturación lineal con el estado inicial de abajo, muestre su estado luego de cada operación, informando cuáles cubetas leyó y cuáles escribió para efectuar: alta 318, alta 661, baja 246.

	Desborde	R1	R2
0	N		
1	N	523	694
2	N	668	

3	N	741	246
4	N	814	229
5	N	734	
6	N		
7	N		
8	N	215	

Considere el mismo estado inicial y repita las operaciones para dispersión doble, usando como segunda función  $f_2(c) = c \bmod 5 + 1$ .

3. Dado el siguiente archivo con saturación lineal encadenada, dibuje los estados sucesivos para las siguientes operaciones: +23, +56, +36, +25. Al finalizar calcule la densidad de empaquetamiento. Para cada operación indique las ranuras leídas y escritas y dibuje el estado siguiente completo.

	Estado	Registro	Enlace
0	O	22	-1
1	O	34	-1
2	O	46	-1
3	V		-1
4	O	59	-1
5	V		-1
6	V		-1
7	O	40	-1
8	V		-1
9	V		-1
10	V		-1

4. Dado el estado de un archivo directo con cubetas para a lo sumo dos registros y con dispersión doble que se detalla, muestre los estados sucesivos completos al realizar la siguiente secuencia de operaciones: +197, +287. La segunda función de dispersión es  $f_2(c) = c \bmod 5 + 1$ . Para cada operación indique la secuencia de cubetas leídas y escritas.

	Desborde	R1	R2
0	N	203	49
1	N	78	155
2	N	65	

3	N	339	
4	N	221	158
5	N	54	
6	N	412	

Considere el mismo estado inicial y repita las operaciones para saturación lineal.

5. Archivo Directo con Dispersión Extensible y Cubetas con Capacidad para 4 Registros:

Tabla:

1 2 3 0

Cubetas Libres: -

Bits de dispersión y claves de registros en cubetas:

0: 2 | 595 351 219 695

1: 2 | 932 368 956 100

2: 2 | 533

3: 2 | 946 178 830

Indicar qué cubetas se leen y escriben y mostrar los estados sucesivos para las siguientes operaciones: alta 51, baja 533, alta 84, baja 368.

6. Dado el archivo directo con dispersión extensible y cubetas con capacidad para 4 registros cuyo estado se muestra a continuación:

Tabla:

2 0 1 3

Cubetas Libres: -

Bits de dispersión y claves de registros en cubetas:

0: 2 | 761 953 909 285

1: 2 | 326 378 566 754

2: 2 | 384

3: 2 | 867 923 895

Grafique los estados sucesivos e informe el costo en lecturas y escrituras para las operaciones: alta 241, alta 406, baja 284.

7. Dado el siguiente estado de un archivo directo con dispersión extensible de cubetas con capacidad para 4 registros

Tabla:

1 3 1 0 1 2 1 0

Cubetas Libres: -

0: 2 | 999 779 63

1: 1 | 124 504 404 492

2: 3 | 461

3: 3 | 153 393 817 585

Muestre la secuencia de estados al realizar las siguientes operaciones, indicando los bloques leídos y escritos: alta 400, baja 461, y bajas de 504 y 400.

8. Dado el archivo directo con dispersión extensible y cubetas con capacidad para 3 registros de abajo, mostrar la secuencia de estados correspondiente a las operaciones: baja 46232, alta 23726, baja 19177, alta 39657. Reporte para cada operación las cubetas que se leen y escriben.

Tabla:

4 0 1 3 2 0 1 3

Cubetas Libres: -

0: 2 | 19177

1: 2 | 16826, 29542, 14514

2: 3 | 62404, 14764

3: 2 | 55847, 24963, 18207

4: 3 | 4623

9. Dado el estado de un archivo directo con cubetas para a lo sumo 4 registros y con dispersión extensible que se detalla, muestre los estados sucesivos **completos** al realizar la siguiente secuencia de operaciones, indicando las cubetas leídas y escritas en cada operación: +716, +232, -798.

Tabla:

1 2 1 3 1 2 1 0

Cubetas Libres: -

0: 3 | 951 743 527

1: 1 | 136 300 832 798

2: 2 | 401 885

3: 3 | 635 475