

Solución:

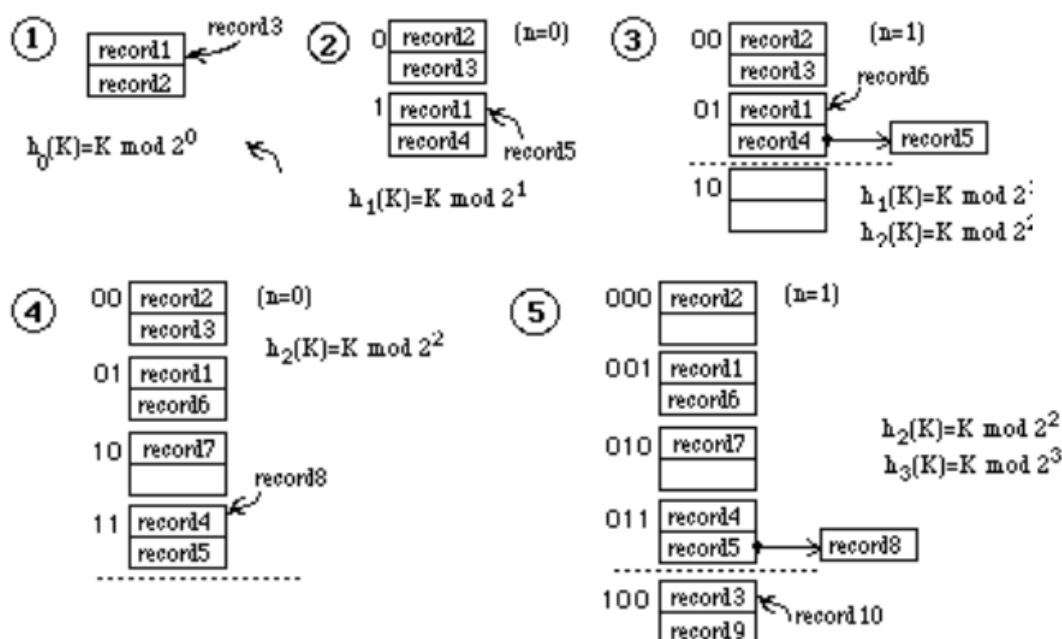
Actividad 3: Técnicas de dispersión. Direccionamiento calculado lineal

1- Un fichero COMPONENTS con *Numcomp* como clave de **direccionamiento calculado** contiene registros con los siguientes valores de Numcomp: 2369, 3760, 4692, 4871, 5659, 1821, 1074, 7115, 1620, 2428, 3943, 4750, 6975, 4981, 9208.

Cargar estos registros en ficheros con direccionamiento **calculado lineal**. Empezar con un solo bloque de disco, utilizando la función hash $h_0 = K \bmod 2^0$, y mostrar cómo crece el fichero y cómo cambian las funciones hash a medida que se insertan registros.

Asumir que los bloques se dividen siempre que se produce un desbordamiento, y mostrar el valor n en cada etapa.

Respuesta:



Inicialmente:

- el tamaño del fichero es $M = 1$ bloque
- el siguiente bloque a expandir es $n = 0$
- $h_0 = K \bmod 2^0$ es la función de dispersión para asignar bucket a un nuevo registro
- $h_1 = K \bmod 2^1$ es la función de expansión para reasignar registros en el momento de la expansión

FIGURAS 1 y 2:

$$h_0(\text{record1}) = 2369 \bmod 2^0 = 0$$

$$h_0(\text{record2}) = 3760 \bmod 2^0 = 0$$

$h_0(\text{record3}) = 4692 \bmod 2^0 = 0$ OVERFLOW \Rightarrow dividir el cubo señalado por n (es decir, el cubo 0) mediante $h_1(K)$ creando el cubo 1 $\Rightarrow n++ \Rightarrow n=1$

$$h_1(\text{record1}) = 2369 \bmod 2^1 = 1$$

$$h_1(\text{record2}) = 3760 \bmod 2^1 = 0$$

$$h_1(\text{record3}) = 4692 \bmod 2^1 = 0$$

Como $n = M \Rightarrow$ Se ha aplicado h_1 a todo el fichero (formado por 1 único bloque, en este caso) \Rightarrow **EXPANSIÓN COMPLETA** \Rightarrow

- el nuevo tamaño del fichero es $M = 2$ bloques
- el siguiente bloque a expandir es $n = 0$
- $h_1 = K \bmod 2^1$ es la función de dispersión para asignar bucket a un nuevo registro
- $h_2 = K \bmod 2^2$ es la función de expansión para reasignar registros en el momento de la expansión

FIGURAS 2 y 3:

$h_1(\text{record4}) = 4871 \bmod 2^1 = 1$

$h_1(\text{record5}) = 5659 \bmod 2^1 = 1$ OVERFLOW \Rightarrow dividir el cubo 0 mediante $h_2(K)$ creando el cubo2 $\Rightarrow n++ \Rightarrow n=1$

$h_2(\text{record2}) = 3760 \bmod 2^2 = 0$

$h_2(\text{record3}) = 4692 \bmod 2^2 = 0$

$h_1(\text{record6}) = 1821 \bmod 2^1 = 1$ OVERFLOW \Rightarrow dividir el cubo 1 mediante $h_2(K)$ creando el cubo 3 $\Rightarrow n++ \Rightarrow n=2$

$h_2(\text{record1}) = 2369 \bmod 2^2 = 1$

$h_2(\text{record4}) = 4871 \bmod 2^2 = 3$

$h_2(\text{record5}) = 5659 \bmod 2^2 = 3$

$h_2(\text{record6}) = 1821 \bmod 2^2 = 1$

Como $n = M \Rightarrow$ Se ha aplicado h_2 a todo el fichero (formado por dos bloques 0 y 1) \Rightarrow

EXPANSIÓN COMPLETA \Rightarrow

- el nuevo tamaño del fichero es $M = 4$ bloques
- el siguiente bloque a expandir es $n = 0$
- $h_2 = K \bmod 2^2$ es la función de dispersión para asignar bucket a un nuevo registro
- $h_3 = K \bmod 2^3$ es la función de expansión para reasignar registros en el momento de la expansión

FIGURAS 4 y 5:

$h_2(\text{record7}) = 1074 \bmod 2^2 = 2$

$h_2(\text{record8}) = 7115 \bmod 2^2 = 3$ OVERFLOW \Rightarrow dividir el cubo 0 mediante $h_3(K)$ creando el cubo4 $\Rightarrow n++ \Rightarrow n=1$

$h_3(\text{record2}) = 3760 \bmod 2^3 = 0$

$h_3(\text{record3}) = 4692 \bmod 2^3 = 4$

$h_2(\text{record9}) = 1620 \bmod 2^2 = 0 \Rightarrow$ como $0 < n=1 \Rightarrow$ el cubo 0 ya está desdoblado \Rightarrow debe aplicarse la función $h_3(K)$

$h_3(\text{record9}) = 1620 \bmod 2^3 = 4$

$h_2(\text{record10}) = 2428 \bmod 2^2 = 0 \Rightarrow$ como $0 < n=1 \Rightarrow$ el cubo 0 ya está desdoblado \Rightarrow debe aplicarse la función $h_3(K)$

$h_3(\text{record10}) = 2428 \bmod 2^3 = 4$ OVERFLOW \Rightarrow dividir el cubo 1 mediante $h_3(K)$

creando el cubo 5 $\Rightarrow n++ \Rightarrow n=2$

$h_3(\text{record1}) = 2369 \bmod 2^3 = 1$

$h_3(\text{record6}) = 1821 \bmod 2^3 = 5$

FIGURAS 6 y 7:

$h_2(\text{record11}) = 3943 \bmod 2^2 = 3$ OVERFLOW \Rightarrow dividir el cubo 2 mediante $h_3(K)$ creando el cubo 6 $\Rightarrow n++ \Rightarrow n=3$

$h_3(\text{record7}) = 1074 \bmod 2^3 = 2$

$h_2(\text{record12}) = 4750 \bmod 2^2 = 2 \Rightarrow$ como $2 < n=3 \Rightarrow$ el cubo 2 ya está desdoblado \Rightarrow debe aplicarse la función $h_3(K) = K \bmod 2^3$

$h_3(\text{record12}) = 4750 \bmod 2^3 = 6$

$h_2(\text{record13}) = 6975 \bmod 2^2 = 3$ OVERFLOW \Rightarrow dividir el cubo 3 mediante $h_3(K)$ creando el cubo 7 $\Rightarrow n++ \Rightarrow n=4$

$h_3(\text{record4}) = 4871 \bmod 2^3 = 7$

$h_3(\text{record5}) = 5659 \bmod 2^3 = 3$

$h_3(\text{record8}) = 7115 \bmod 2^3 = 3$

$h_3(\text{record11}) = 3943 \bmod 2^3 = 7$

$h_3(\text{record13}) = 6975 \bmod 2^3 = 7$ (está en overflow, por lo que se mantiene en la lista de desbordamiento, pero **en este caso NO se hace expansión de cubos**)

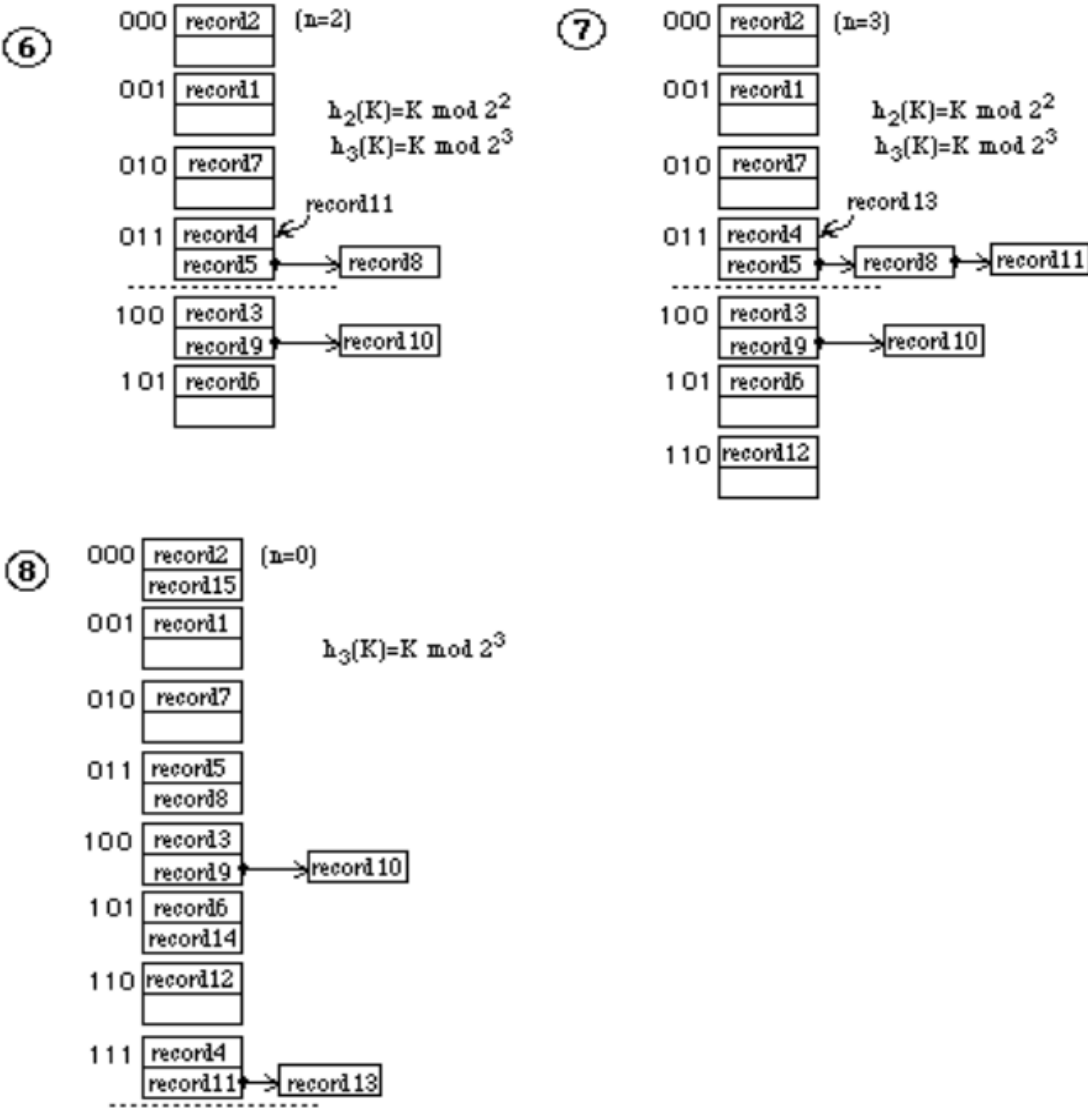
Como $n = M \Rightarrow$ Se ha aplicado h_3 a todo el fichero (formado por cuatro bloques) \Rightarrow

EXPANSIÓN COMPLETA \Rightarrow

- el nuevo tamaño del fichero es $M = 8$ bloques
- el siguiente bloque a expandir es $n = 0$
- $h_3 = K \bmod 2^3$ es la función de dispersión para asignar bucket a un nuevo registro
- $h_4 = K \bmod 2^4$ es la función de expansión para reasignar registros en el momento de la expansión

FIGURA 8:

$h_3(\text{record14}) = 4981 \bmod 2^3 = 5$
 $h_3(\text{record15}) = 9208 \bmod 2^3 = 0$



ALTERNATIVAS A LA INSERCIÓN DE REGISTROS: FACTOR DE CARGA

Es más común establecer una política de desdoblamiento que especifica un **factor de carga del fichero** que dispare el desdoblamiento del bucket señalado por **n** (generalmente un porcentaje de carga de los buckets), en vez de realizar el desdoble cada vez que se produce una situación de overflow, como en el ejemplo anterior.

El *factor de carga l* se define como $l = r / (bfr * N)$, donde **r** es el número actual de registros del fichero, **bfr** es el número máximo de registros que pueden encajar en un cubo, y **N** es el número actual de cubos del fichero. Las divisiones se pueden lanzar cuando **l** es mayor de un cierto umbral (p. ej, 0.9).

Por ejemplo, si el factor de bloqueo es 2 (como en el ejemplo anterior), la expansión de los bloques se lleva a cabo del siguiente modo:

Cuando $N=1, l = r / (bfr * N) < 0,9 \Rightarrow r > 1,8 \Rightarrow$ desdoblar tras incluir el registro 2

Cuando $N=2, l = r / (bfr * N) < 0,9 \Rightarrow r > 3,6 \Rightarrow$ desdoblar tras incluir el registro 4

Cuando $N=3, l = r / (bfr * N) < 0,9 \Rightarrow r > 5,4 \Rightarrow$ desdoblar tras incluir el registro 6

...

ELIMINACIÓN DE REGISTROS

En caso de la **eliminación de registros**, es posible combinar bloques en orden inverso a como fueron creados. Las combinaciones se pueden lanzar cuando la carga cae por debajo de otro umbral (p. ej., 0.7).

REALIZACIÓN DE BÚSQUEDAS

Para analizar las ventajas de este modelo, a continuación se presenta el modo en que se realizan las búsquedas de registros.

Algoritmo: *if* **n** = 0 *then* $v = h_j(K)$
 else {
 $v = h_j(K)$
 if **v** < **n** *then* $v = h_{j+1}(K)$
 }

buscar en el cubo con valor de dispersión v (y su desbordamiento, si lo hay)

Ejemplo: Los ejemplos se basan en la situación mostrada en la Figura 7, donde existían 12 registros en el fichero. En este caso:

- $n = 3$
- la función de dispersión es $h_2(k) = k \bmod 2^2$
- la función de expansión es $h_3(k) = k \bmod 2^3$

Ej: *búsqueda de r8 (7715)*

$n \neq 0 \Rightarrow v = h_2(r8) = 7115 \bmod 2^2 = 3$

3 no es menor que n \Rightarrow buscar en el cubo 3 (y su desbordamiento, si lo hay)

Ej: *búsqueda de r12 (4750)*

$n \neq 0 \Rightarrow v = h_2(4750) = 4750 \bmod 2^2 = 2$

2 es menor que n $\Rightarrow v = h_3(r12) = 4750 \bmod 2^3 = 6$

buscar en cubo 6 (y su desbordamiento, si lo hay)