

función de dispersión

- Da Df donde se almacena clave.

registro - suónima

- 2 datos dm = df

deborde

- Cuando no hay + esp. en el nodo. si hay deb. hay coll.

clave intrusa

- Clave q' está en una direl. q' no es la asignada a fin de hashing.

Soluciones de Df.

Sanctus Progrein

→ Busca cantidad de datos que
ir un con espais

100, 100 x 100 (salvo si hay 0)

$$f(2) = 3 \text{ c. y } 0 \rightarrow \text{inicio}$$

$$f(3) = 5 \text{ c. y } 0 \rightarrow \text{inicio}$$

$$f(8) = 4$$

$$f(14) = 3 \text{ L } 3, 4, 5, 6, 7$$

$$f(n) = x \bmod n$$

direl	reg	reg
0	22	
1	78	
2	56	
3	58	91
4	25	81
5	60	27
6	38	14
7		
8	85	
9		
10	76	21

$$f(5) = 3$$

$$f(8) = 4$$

$$f(60) = 5$$

$$f(3) = 4$$

$$f(9) = 3 \text{ c.}$$

$$f(2) = 5 \text{ c.}$$

Ordenar de ene.

$$\frac{1.5}{11.2}$$

[MAYAS DE BORRADO NO SE BORRAN, NADA NO ESTÁ LLENO]

Solo se pone si no hay. Pasa info o marca de borrado

Sanctus Progrein

0			
1	2	78	
2	-1	56	
3	7	58	
4	-1	92	
5	-1	60	
6	-1	72	
7	-1	25	
8	-1	85	
9	-1		
10			

$$53 = 3$$

$$20 = 1$$

$$60 = 5$$

$$92 = 4$$

$$45 = 3$$

$$56 = 2 \rightarrow \text{deborde}$$

$$21 = 3 \rightarrow \text{deborde}$$

$$82 = 6 \rightarrow \text{re q' 23 es intruso, lo marca a por supueste + problema.}$$

Cambio de

23 = 1 en un caso plata → Al agregar símbolo 1 el q' referencia

23 = 1, 1, 9, 6, 2

→ 23 = 1, 1, 9, 6, 2 la por 9, 6, 2 el nuevo referencia q' tenía en base.

→ 23 = 1, 1, 9, 6, 2 q' está en 2 y lo dir. en 2

Al borrar dato de

Hay q' actualizar conexiones del elem. previo. Aplica hashing

direl bote, borra de 10

Sobre clave bote de finar 74000000

q' tiene y pongolo q'

L6, L8, E+, L3, E3 (reducir), E6 (nuevo dato)

apuntador.

hashing

ARRO DE DEBORDE

0	-1		
1	1	78	56
2	-1		
3	-1	58	
4	-1	92	
5	-1	60	
6	-1		
7	-1		
8	-1	85	
9	-1		
10			

D	1	2
1	6	23
2		

$$f(2) = 1$$

$$f(3) = 1$$

→ 23 = 1, 1, 9, 6, 2 q' 10 referencia.

4	1	92	
5	1	60	
6	1		
7	1		
8	1	85	
9	1		
10	1		

-12 = Asumiendo el valor de 10 referencias.

Al buscar donde viene el 1, si el valor ya tiene ref. a donde debe ir.

La idea y buscar es que si el valor ya tiene ref. a donde debe ir.

De bit displacement

Si los bits están en la posición de los bits, los bits están en la posición de los bits.

0			
1	34	78	
2			
3	91		
4	12		
5	60		
6	89		
7	36		
8	85		
9			
10			

24=1 12=3 (valor 4)

28=1 39=5 (A)

60=5 89=1 (2)

85=9 36=3 (valor 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100)

011=3 12=5 (no m. 4)

Se debe borrar siempre