

Estructuras de Datos

Hash

Definiciones - Colisiones

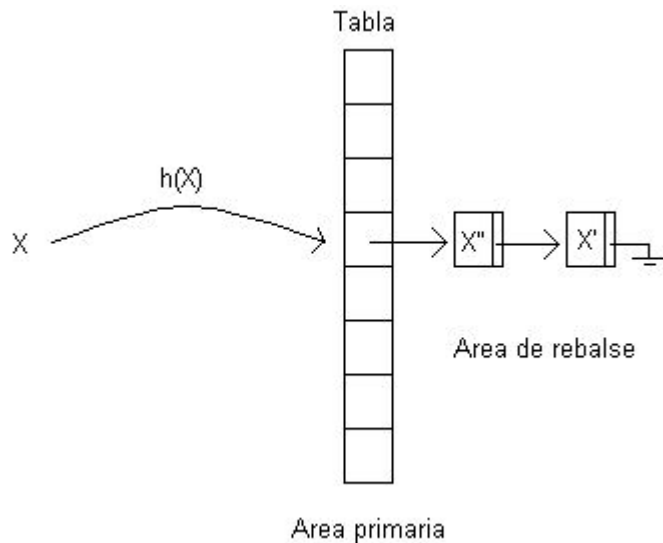
Para resolver el problema de las colisiones, existen dos grandes familias de métodos:

- Encadenamiento (usar estructuras dinámicas).
- Direccionamiento abierto (intentar con otra función de hash).

En esta presentación vamos a analizar el Encadenamiento.

Colisiones - Encadenamiento

La idea de este método es que todos los elementos que caen en el mismo casillero se almacenen en una estructura auxiliar. En este ejemplo podemos ver que se usa una LSE en la cual se realiza una búsqueda secuencial.





Colisiones - Encadenamiento - Hashing con listas mezcladas

Esta es una variante de Encadenamiento donde en lugar de usar un área de rebalse los elementos se almacenan en cualquier lugar libre del área primaria. Para poder saber dónde buscar se utiliza un puntero al siguiente elemento, es decir, construimos listas dentro de la tabla hash.

Veamos un ejemplo a continuación.



Colisiones - Encadenamiento - Hashing con listas mezcladas

Supongamos que tenemos una tabla hash con $m=10$. Nuestra función de hash es simplemente:

$$h(X) = X \bmod 10$$

Usamos Hashing con listas mezcladas para resolver colisiones.



Colisiones - Encadenamiento - Hashing con listas mezcladas

Inicialmente tenemos la tabla hash vacía.


Podemos ver que tenemos un espacio para guardar el dato y un espacio para el puntero al siguiente elemento.

0		
1		
2		
3		
4		
5		
6		
7		
8		
9		



Colisiones - Encadenamiento - Hashing con listas mezcladas

Si almacenamos el 25, podemos ver que, usando la función hash, se ocuparía la clave 5 en la tabla por lo que tendríamos:

0		
1		
2		
3		
4		
5	25	
6		
7		
8		
9		



Colisiones - Encadenamiento - Hashing con listas mezcladas

Ahora se almacena el 39 usando la función hash, se ocupa la clave 9 en la tabla por lo que tendríamos:

0		
1		
2		
3		
4		
5	25	└─┐
6		
7		
8		
9	39	└─┐



Colisiones - Encadenamiento - Hashing con listas mezcladas

Se guarda el 36, usando la función hash la clave es 6. La tabla queda:

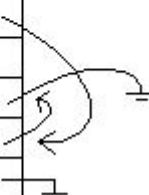
0		
1		
2		
3		
4		
5	25	└─┴─┘
6	36	└─┴─┘
7		
8		
9	39	└─┴─┘



Colisiones - Encadenamiento - Hashing con listas mezcladas

Un tiempo después miramos la tabla y lo que vemos es:

0		
1		
2		
3		
4		
5	25	
6		
7	48	
8	15	
9	39	





Colisiones - Encadenamiento - Hashing con listas mezcladas

Analizándola podemos concluir que:

- el 15 está almacenado en una clave que no le corresponde porque su clave estaba ocupada (25), seguramente en el momento de guardarla las claves sucesivas también lo estaban y, la primera que encontró libre fue la clave 8;
- el 48, al querer guardarse encontró su clave ocupada y, la primer clave que encontró libre fue la 7.

0		
1		
2		
3		
4		
5	25	
6		
7	48	
8	15	
9	39	

Colisiones - Encadenamiento

Ahora bien, ¿cuándo conviene usar un área de rebalse y, cuándo usar listas mezcladas?

No hay una única respuesta sino que en realidad depende del problema pero, en general, esta variante es útil cuando la tabla no está densamente poblada.

Para tener una idea de este concepto vamos a definir factor de carga de una Tabla hash.

Colisiones - Encadenamiento

El grado de ocupación de una tabla hash se determina mediante el factor de carga, que es la fracción ocupada de la tabla y es un número que está entre 0 y 1 si está vacía o llena respectivamente.

Dicho factor se obtiene dividiendo la cantidad de elementos almacenados sobre el total de espacio disponible. Es decir, si llamamos α a dicho factor, hay n elementos almacenados en una tabla hash de tamaño m entonces:

$$\alpha = \frac{n}{m}$$

Se puede verificar que el costo de búsqueda sólo depende del factor de carga α , y no del tamaño de la tabla.

Colisiones - Encadenamiento - Eliminación

El caso de eliminar con un área de rebalse es sencillo: sólo hay que eliminar el elemento allí, por lo que la complejidad viene dada por el costo de eliminarla de la misma.

Pero, en el caso del hashing con las lista mezcladas, el algoritmo es más complejo, ya que debe reenlazar las listas. Pensemos en el ejemplo anterior: ¿qué pasaría si elimino el 15? En ese caso deberíamos reenlazar el 48 ya que de otra forma lo perderíamos.

En la siguiente presentación vamos a ver el otro enfoque: Direcccionamiento Abierto.