INSERT, SEARCH, y DELETE. Por ejemplo, un compilador para un lenguaje de programación

mantiene una tabla de símbolos, en el que las claves de los elementos son cadenas arbitrarias de caracteres que

corresponden a los identificadores en el lenguaje. Una tabla hash es una estructura de datos eficaz para

la aplicación de los diccionarios. Aunque la búsqueda de un elemento en una tabla hash puede tardar

como la búsqueda de un elemento en una lista vinculada, Θ (n) en el peor de los casos-en la práctica, hashing

funciona extremadamente bien. Bajo supuestos razonables, el tiempo de espera para buscar un

elemento en una tabla hash es O (1).

Una tabla hash es una generalización de la noción simple de una matriz común. Dirigiéndose directamente

en una matriz común hace uso efectivo de nuestra capacidad para analizar una posición arbitraria en un

matriz en O (1) hora. La sección 11.1 describe el direccionamiento directo con más detalle. direccionamiento directo

es aplicable cuando podemos darnos el lujo de asignar una matriz que tiene una posición para cada posible

clave.

Direccionamiento directo es una técnica sencilla que funciona bien cuando el universo U de claves que se

razonablemente pequeño. Supongamos que una aplicación necesita un conjunto dinámico en el que cada elemento tiene

una clave extraídos de la U = universo {0, 1, ..., m - 1}, donde m no es demasiado grande. Vamos a

asumir que no hay dos elementos tienen la misma clave.

Para representar el conjunto dinámico, utilizamos una matriz o directa dirección de la tabla, que se denota por T [0? m -

1], en el que cada posición, o en la ranura, corresponde a una clave en el universo U. La figura 11.1

ilustra el enfoque, los puntos de la ranura k de un elemento en el conjunto con la clave k. Si el conjunto no contiene

elemento con clave k, entonces T [k] = NIL.



La operación del diccionario son triviales de implementar.  
DIRECTA de búsqueda de direcciones (T, k)  
retorno T [k]  
DIRECTA DE DIRECCIÓN-INSERT (T, x)  
T [tecla [x]] ← x  
DIRECTA DE DIRECCIÓN-delete (T, x)  
T [tecla [x]] ← NIL

Colisión de resolución por el encadenamiento de  
En el encadenamiento, ponemos todos los elementos que hash de la misma ranura en una lista enlazada, como se muestra en  
Figura 11.3. J ranura contiene un puntero a la cabeza de la lista de todos los elementos almacenados que hash de j;  
si no hay tales elementos, ranura j contiene NIL.

Escuchar

Leer fonéticamente

Diccionario



Figura 11.3: resolución de colisiones por encadenamiento. Cada ranura tabla hash T [j] contiene una lista enlazada  
de todas las claves cuyo valor hash es j. Por ejemplo, h (k1) = h (K4) y h (K5) = h (k2) = h (K7).  
La operación del diccionario en una tabla hash T son fáciles de implementar cuando las colisiones son  
resuelta por encadenamiento.  
ENCADENADO-HASH-INSERT (T, x)  
Márquese con una X a la cabeza de la lista T [h (clave [x])]  
ENCADENADO-HASH-SEARCH (T, k)  
búsqueda de un elemento con clave k en la lista T [h (k)]  
ENCADENADO-HASH-delete (T, x)  
x borrar de la lista T [h (clave [x])]