



# Escuela Superior Politécnica de Chimborazo

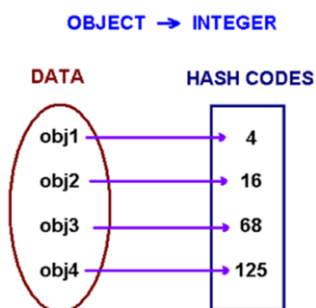
Facultad de Informática y Electrónica  
Escuela de Ingeniería en Sistemas



## Estructura De Datos

### Trabajo Grupal

Tema: Exposición de Hashing.



#### Integrantes:

Juana Delgado

Aldas Wilmer

Cando Lizbeth

Pilataxi Claudio

Romero Marcelo

#### Semestre:

Tercero "A"

#### Fecha de Envío:

14/07/2016

#### Códigos:

6078

6035

5951

6042

6030

#### Año Lectivo:

Abril-Agosto 2016

#### Fecha de Recepción:

21/07/2016

Riobamba-Ecuador

## ○ Introducción

Cuando el número de claves que se almacenan efectivamente es pequeño comparado con el número total de claves posibles, una tabla hash ofrece una alternativa eficiente a una tabla de búsqueda directa, utilizando solamente un arreglo de tamaño proporcional a la cantidad de claves almacenadas. En lugar de utilizar la clave como un índice directamente, el índice se calcula a partir de la clave utilizando una función hash.

## ○ Objetivos

Conocer el significado de hashing.

Solucionar problemas relacionados con tablas hash.

Conocer los métodos para resolver los problemas.

Conocer las dificultades que aparezcan en los ejercicios y dar solución a los mismos.

## ○ Marco teórico

Una tabla hash, matriz asociativa, mapa hash, tabla de dispersión o tabla fragmentada es una estructura de datos que asocia llaves o claves con valores. La operación principal que soporta de manera eficiente es la búsqueda: permite el acceso a los elementos (teléfono y dirección, por ejemplo) almacenados a partir de una clave generada (usando el nombre o número de cuenta, por ejemplo). Funciona transformando la clave con una función hash en un hash, un número que identifica la posición (casilla o cubeta) donde la tabla hash localiza el valor deseado.

Un ejemplo práctico para ilustrar que es una tabla hash es el siguiente: Se necesita organizar los periódicos que llegan diariamente de tal forma que se puedan ubicar de forma rápida, entonces se hace de la siguiente forma - se hace una gran caja para guardar todos los periódicos (una tabla), y se divide en 31 contenedores (ahora es una "hash table" o tabla fragmentada), y la clave para guardar los periódicos es el día de publicación (índice). Cuando se requiere buscar un periódico se busca por el día que fue publicado y así se sabe en qué zócalo (bucket) está. Varios periódicos quedarán guardados en el mismo zócalo (es decir colisionan al ser almacenados), lo que implica buscar en la sub-lista que se guarda en cada zócalo. De esta forma se reduce el tamaño de las búsquedas de  $O(n)$  a  $O(\log(n))$ .

Las tablas hash se suelen implementar sobre vectores de una dimensión, aunque se pueden hacer implementaciones multi-dimensionales basadas en varias claves. Como en el caso de los arrays, las tablas hash proveen tiempo constante de búsqueda promedio  $O(1)$ , sin importar el número de elementos en la tabla. Para tablas hash con resolución de colisiones de hashing abierto (si la tabla no está balanceada) queda  $O(n)$ , es decir, en función del número de elementos.

Comparada con otras estructuras de arrays asociadas, las tablas hash son más útiles cuando se almacenan grandes cantidades de información.

### ○ Ejercicios y resolución

## INSERCIÓN

Utilizando el método de encadenamiento enlazado

Ejercicio # 01:

Insertar {30, 15, 20, 7, 23, 49, 10, 37}

$$30 \text{ MOD } 7 = 2$$

INDICE	VALOR
0	
1	
2	30
3	
4	
5	
6	
7	

$$15 \text{ MOD } 7 = 1$$

INDICE	VALOR
0	
1	15
2	30
3	
4	
5	
6	
7	

$$20 \text{ MOD } 7 = 6$$

INDICE	VALOR
0	
1	15
2	30
3	
4	
5	
6	20
7	

$$7 \text{ MOD } 7 = 0$$

INDICE	VALOR
0	7
1	15
2	30
3	
4	
5	
6	20
7	

$$10 \text{ MOD } 7 = 3$$

INDICE	VALOR
0	7
1	15
2	30
3	10
4	
5	
6	20
7	

$$23 \text{ MOD } 7 = 2$$

INDICE	VALOR
0	7
1	15
2	30
3	
4	
5	
6	20
7	

$$37 \text{ MOD } 7 = 2$$

INDICE	VALOR
0	7
1	15
2	30
3	10
4	
5	
6	20
7	

$$49 \text{ MOD } 7 = 0$$

INDICE	VALOR
0	7
1	15
2	30
3	
4	
5	
6	20
7	

### Ejercicio # 02:

Insertar {13, 22, 38, 45, 25,54}

$$13 \text{ MOD } 6 = 1$$

INDICE	VALOR
0	
1	13
2	
3	
4	
5	

$$22 \text{ MOD } 6 = 4$$

INDICE	VALOR
0	
1	13
2	
3	
4	22
5	

$$38 \text{ MOD } 6 = 2$$

INDICE	VALOR
0	
1	13
2	38
3	
4	22
5	

$$45 \text{ MOD } 6 = 3$$

$$25 \text{ MOD } 6 = 1$$

$$10 \text{ MOD } 6 = 4$$

INDICE	VALOR
0	
1	13
2	38
3	45
4	22
5	

INDICE	VALOR
0	
1	13
2	38
3	45
4	22
5	

INDICE	VALOR
0	
1	13
2	38
3	45
4	22
5	

## INSERCIÓN

Utilizando el método de exploración

### Ejercicio 1

Insertar {8,12,14,7,11,13,99,69}

Insertar el 8  
 $8 \text{ mod } 7 = 1$

Pos.	Elem.
0	
1	8
2	
3	
4	
5	
6	
7	

Insertar el 12  
 $12 \text{ mod } 7 = 5$

Pos.	Elem.
0	
1	8
2	
3	
4	
5	12
6	
7	

Insertar el 14  
 $14 \text{ mod } 7 = 0$

Pos.	Elem.
0	14
1	8
2	
3	
4	
5	12
6	
7	

Insertar el 7  
 $7 \text{ mod } 7 = 0$

Se produce una **COLISIÓN** ya que en ese índice ya existe un elemento

Se busca hacia adelante una posición vacía

En este caso en el índice 2

Pos.	Elem.
0	14
1	8
2	7
3	
4	
5	12
6	
7	

Insertar el 11  
 $11 \text{ mod } 7 = 4$

Pos.	Elem.
0	14
1	8
2	7
3	
4	11
5	12
6	
7	

Insertar el 13  
 $13 \bmod 7 = 6$

Pos.	Elem.
0	14
1	8
2	7
3	
4	11
5	12
6	13
7	

Insertar el 99

$$99 \bmod 7 = 1$$

Se produce una COLISIÓN ya que en ese índice ya existe un elemento

Se busca hacia adelante una posición vacía

En este caso en el índice 3

Pos.	Elem.
0	14
1	8
2	7
3	99
4	11
5	12
6	13
7	

Insertar el 69

$$69 \bmod 7 = 6$$

Se produce una COLISIÓN ya que en ese índice ya existe un elemento

Se busca hacia adelante una posición vacía

En este caso en el índice 7

Pos.	Elem.
0	14
1	8
2	7
3	99
4	11
5	12
6	13
7	69

### RESPUESTA

Pos.	Elem.
0	14
1	8
2	7
3	99
4	11
5	12
6	13
7	69

## Ejercicio 2

Insertar {73,58,48,31,28,10,55,88}

Insertar el 73  
 $73 \bmod 7 = 3$

Pos.	Elem.
0	
1	
2	
3	73
4	
5	
6	
7	

Insertar el 58  
 $58 \bmod 7 = 2$

Pos.	Elem.
0	
1	
2	58
3	73
4	
5	
6	
7	

Insertar el 48  
 $48 \bmod 7 = 6$

Pos.	Elem.
0	
1	
2	58
3	73
4	
5	
6	48
7	

Insertar el 31  
 $31 \bmod 7 = 3$

Se produce una **COLISIÓN** ya que en ese índice ya existe un elemento

Se busca hacia adelante una posición vacía

En este caso en el índice 4

Pos.	Elem.
0	
1	
2	58
3	73
4	31
5	
6	48
7	

Insertar el 28  
 $28 \bmod 7 = 0$

Pos.	Elem.
0	28
1	
2	58
3	73
4	31
5	
6	48
7	

Insertar el 10

$$10 \bmod 7 = 3$$

Se produce una COLISIÓN ya que en ese índice ya existe un elemento

Se busca hacia adelante una posición vacía

En este caso en el índice 5

Pos.	Elem.
0	28
1	
2	58
3	73
4	31
5	10
6	48
7	

Insertar el 55

$$55 \bmod 7 = 6$$

Se produce una COLISIÓN ya que en ese índice ya existe un elemento

Se busca hacia adelante una posición vacía

En este caso en el índice 7

Pos.	Elem.
0	28
1	
2	58
3	73
4	31
5	10
6	48
7	55

#### RESPUESTA

Pos.	Elem.
0	28
1	
2	58
3	73
4	31
5	10
6	48
7	55



Insertar el 88

$$88 \bmod 7 = 4$$

Se produce una COLISIÓN ya que en ese índice ya existe un elemento

Se busca hacia adelante una posición vacía

En este caso en el índice 1

Pos.	Elem.
0	28
1	88
2	58
3	73
4	31
5	10
6	48
7	55

- Ejemplo 1 de inserción utilizando el método de encadenamiento enlazado

Insertar {13, 22, 38, 45, 25, 54}

Buscar el 

Pos.	Elem.
------	-------

 nodo 10

Para buscar el nodo 10 sacamos el modulo con el 6.



$10 \bmod 6 = 4$  Encontrado en la segunda columna.



- Ejemplo 2 de inserción utilizando el método de exploración.

Insertar {73, 58, 48, 31, 28, 10, 55, 88}

Buscar el elemento 31



0	28
1	88
2	58
3	73
4	31
5	10
6	48
7	55

Para buscar el nodo 31 sacamos el modulo con la posición dela tabla.

$$31 \bmod 7 = 3$$

- Preguntamos 73 = 31 NO

Seguimos la búsqueda desde esa posición.

- Preguntamos 31 = 31 Si

**Encontrado**

- **Conclusiones:**
  - Se ha logrado conocer el significado de hashing.
  - Se ha conseguido solucionar problemas relacionados con tablas hash.
  - Se ha conseguido conocer los distintos tipos de tablas y soluciones.
  - Se domina el manejo de tablas hash y su funcionamiento.

- **Recomendaciones:**

Se recomienda usar el método más adecuado para resolver los distintos tipos de problemas que se puedan encontrar con las tablas hash.

Con las tablas hash conseguimos ahorrar tiempo y recursos para la resolución de problemas.