



Universidad  
Rafael Landívar

Tradición Jesuita en Guatemala

# Tablas Hash

Colisiones y aplicaciones

**¿Qué son las tablas Hash?**



**¿Cuál es la función más importante?**



**¿Cuáles son las otras 3 funciones  
necesarias?**



**¿Cuáles son las otras 3 funciones necesarias?**



# Arreglos

5

0

1

2

3

4

5

6

7

8

9

N=10

# Arreglos

0  
1  
2  
3  
4  
5  
6  
7  
8  
9

5

N=10

# Arreglos

23

0	5
1	
2	
3	
4	
5	
6	
7	
8	
9	

N=10



# Arreglos

0

5

1

23

2

3

4

5

6

7

8

9

N=10

# Arreglos

0

5

1

23

2

90

3

75

4

4

5

13

6

27

7

50

8

62

9

N=10

# Arreglos

1

0  
1  
2  
3  
4  
5  
6  
7  
8  
9

5
23
90
75
4
13
27
50
62

N=10

# Arreglos

0

5

1

23

2

90

3

75

4

4

5

13

6

27

7

50

8

62

9

1

**N=10**

# Arreglos

¿Costo de buscar 5?

$O(1)$

0	5
1	23
2	90
3	75
4	4
5	13
6	27
7	50
8	62
9	1

**N=10**

# Arreglos

¿Costo de buscar 1?

$O(10)$

0	5
1	23
2	90
3	75
4	4
5	13
6	27
7	50
8	62
9	1

**N=10**

# Tablas Hash

0

1

2

3

4

5

6

7

8

9

N=10

# Tablas Hash

5 -> hash(key) -> index

0	
1	
2	
3	
4	
5	
6	
7	
8	
9	

N=10



# Tablas Hash

5 -> hash(key) -> 3

0	
1	
2	
3	5
4	
5	
6	
7	
8	
9	

N=10

# Tablas Hash

23 → hash(key) → 7

0	
1	
2	
3	5
4	
5	
6	
7	23
8	
9	

N=10

# Tablas Hash

key -> hash(key) -> index

0	
1	
2	
3	5
4	
5	
6	
7	23
8	
9	

N=10

# Tablas Hash

1 -> hash(key) -> 3

0	95
1	47
2	36
3	5
4	38
5	1
6	4
7	23
8	19
9	50

N=10

# Tablas Hash

¿Costo de buscar 95?

$O(1)$

0	95
1	47
2	36
3	5
4	38
5	1
6	4
7	23
8	19
9	50

**N=10**

# Tablas Hash

¿Costo de buscar 50?

0	95
1	47
2	36
3	5
4	38
5	1
6	4
7	23
8	19
9	50

**N=10**

# Tablas Hash

¿Costo de buscar 50?

$O(1)$

0	95
1	47
2	36
3	5
4	38
5	1
6	4
7	23
8	19
9	50

**N=10**

# Tablas Hash

¿Costo de buscar 50?

$O(1)$

key  $\rightarrow$  hash(key)  $\rightarrow$  index

0	95
1	47
2	36
3	5
4	38
5	1
6	4
7	23
8	19
9	50

**N=10**



# Tablas Hash

7 -> hash(key) -> 3

0	95
1	47
2	36
3	5
4	38
5	1
6	4
7	23
8	19
9	50

N=10

# Colisiones

Las colisiones se producen cuando dos o más objetos (todos diferentes) dan como resultado un mismo índice.

5 -> hash(key) -> 3  
7 -> hash(key) -> 3

0	95
1	47
2	36
3	5
4	38
5	1
6	4
7	23
8	19
9	50

N=10

¿Qué algoritmo es mejor para la  
función hash?  
¿Cómo tratar las colisiones?

# La función de dispersión “hash”

Consiste en un algoritmo que asigna un índice a cada elemento de la tabla mediante una transformación del elemento en cuestión. En informática existen variantes en las funciones hash que transforman el elemento, pero siempre se obtiene el índice en la tabla.

Desafortunadamente no hay manera eficaz de evitar por completo las colisiones en la tabla, sin embargo, si se puede reducir la probabilidad.

key  $\rightarrow$  hash(key)  $\rightarrow$  index

# Restas sucesivas

En este algoritmo la función se emplea con claves numéricas entre las que existen espacios vacíos de tamaño conocido, obteniéndose direcciones consecutivas. Por ejemplo, los alumnos de la universidad Rafael Landívar que entraron en el año 2019, sus números de carnet son consecutivos y está definido el número de alumnos.

1500019 -> hash(key) -> 0

1558619 -> hash(key) -> 586

# División modular

En este algoritmo el índice de un número es el residuo de la división de ese número entre un número N prefijado, preferentemente primo. Los números se guardarán en las direcciones de memoria de 0 a N-1.

$$(5+1) \% 10 \rightarrow \text{hash(key)} \rightarrow 6$$

$$(7+1) \% 10 \rightarrow \text{hash(key)} \rightarrow 8$$

# Mitad del cuadrado

Este algoritmo consiste en elevar al cuadrado la clave y luego seleccionar las cifras centrales del resultado.

$709^2 \rightarrow \text{hash}(\text{key}) \rightarrow 502681$

$879^2 \rightarrow \text{hash}(\text{key}) \rightarrow 772641$



# Socialización de los algoritmos



# Resolución de conflictos

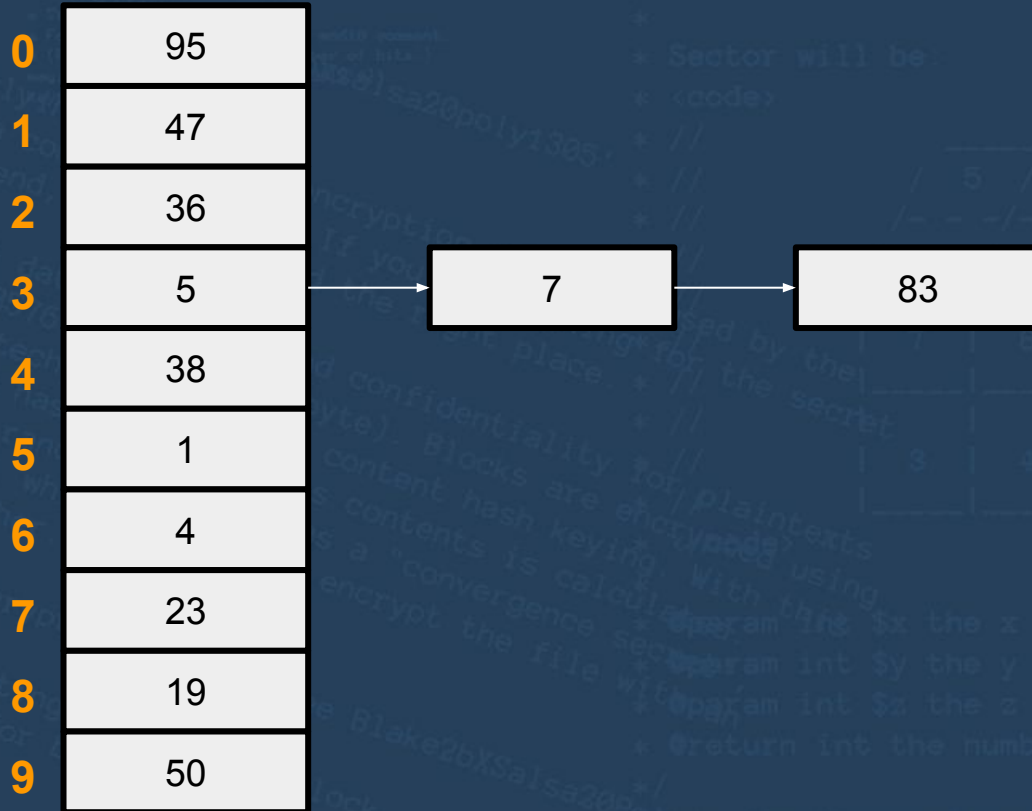
7 -> hash(key) -> 3

0	95
1	47
2	36
3	5
4	38
5	1
6	4
7	23
8	19
9	50

N=10

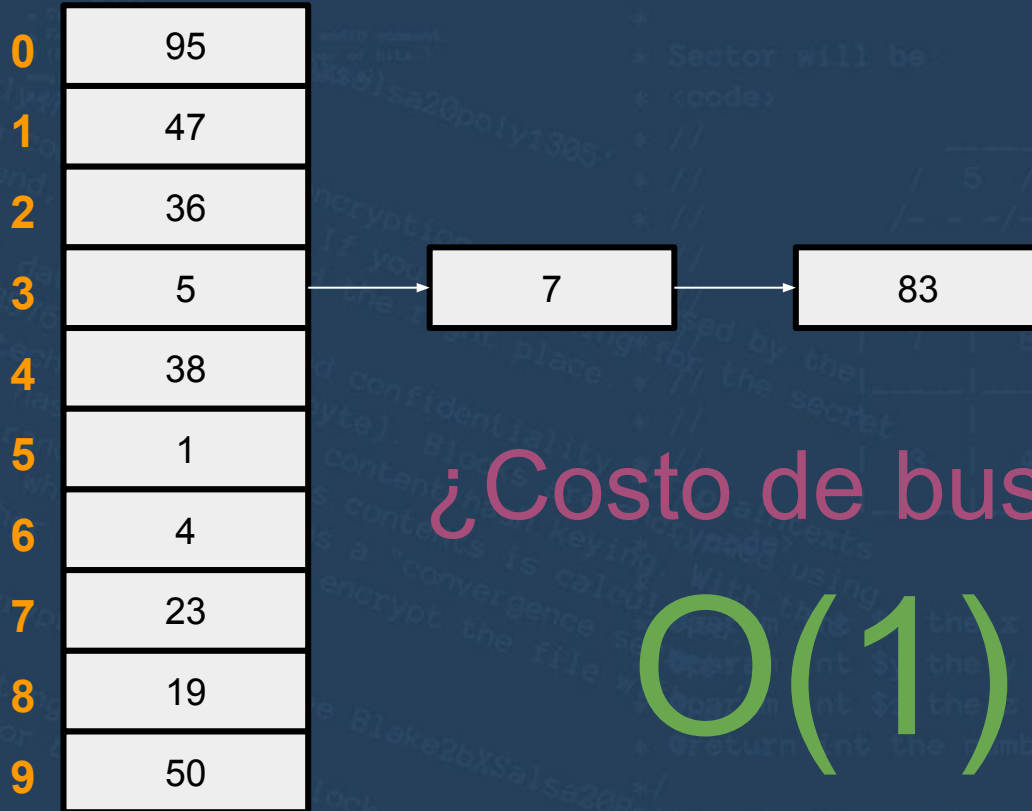
# Listas enlazadas

N=10



# Listas enlazadas

N=10

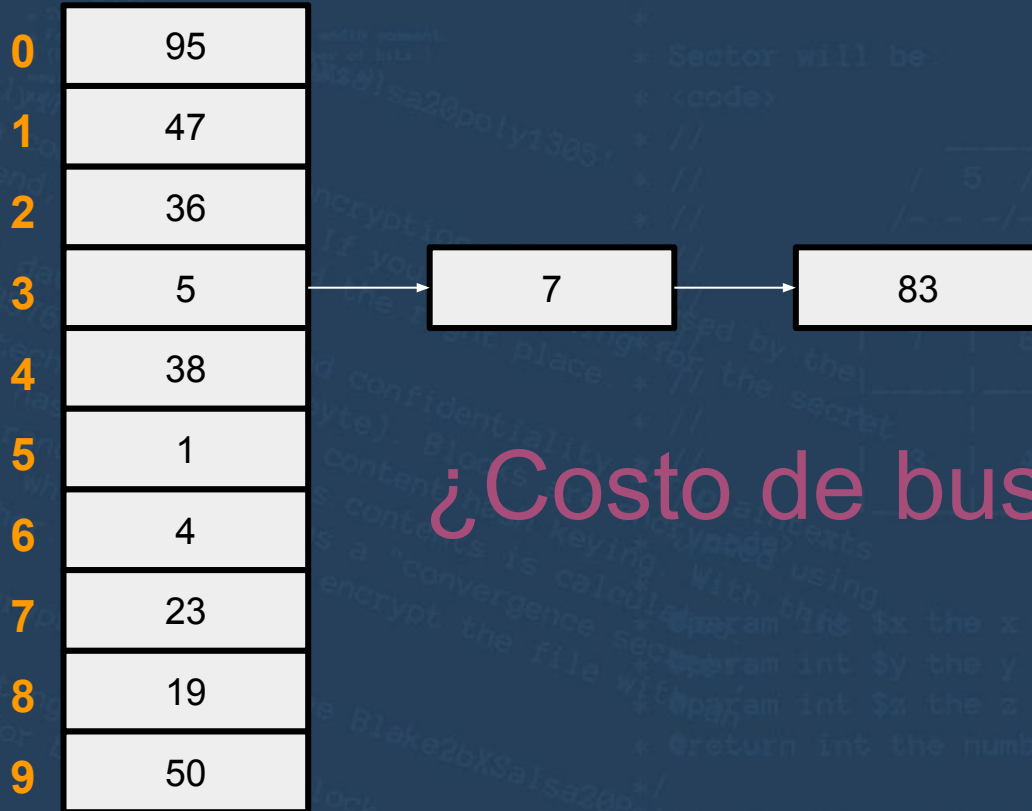


¿Costo de buscar 5?

$O(1)$

# Listas enlazadas

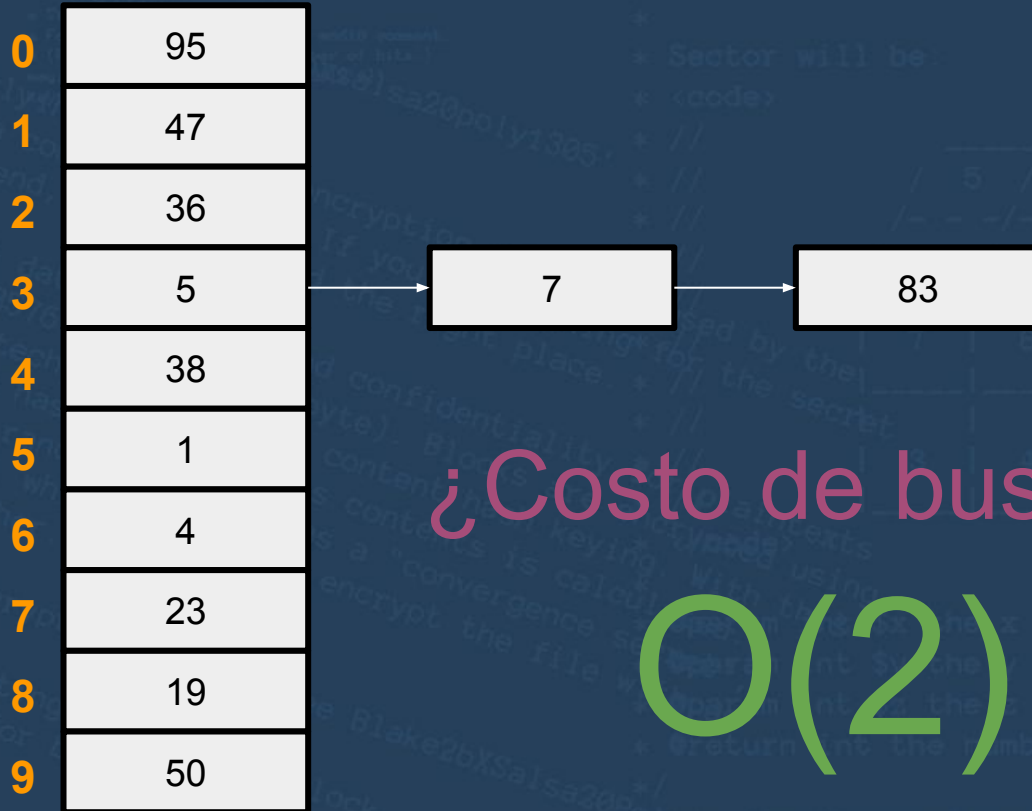
N=10



¿Costo de buscar 7?

# Listas enlazadas

N=10

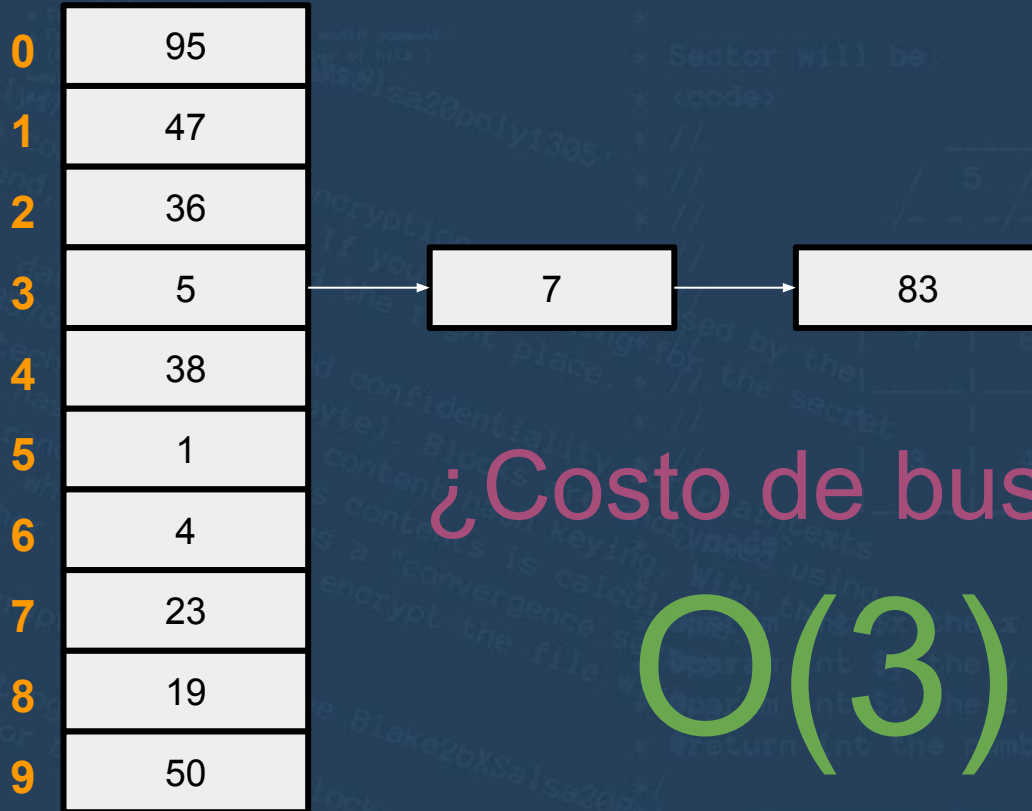


¿Costo de buscar 7?

$O(2)$

# Listas enlazadas

N=10



¿Costo de buscar 83?

$O(3)$

# Mediante exploración

7 -> hash(key) -> 3

0	95
1	47
2	36
3	5
4	38
5	
6	4
7	23
8	19
9	50

N=10

# Mediante exploración

7 -> hash(key) -> 3



0	95
1	47
2	36
3	5
4	38
5	
6	4
7	23
8	19
9	50

N=10



# Mediante exploración

7 -> hash(key) -> 3

0	95
1	47
2	36
3	5
4	38
5	
6	4
7	23
8	19
9	50

N=10

# Mediante exploración

7 -> hash(key) -> 3

0	95
1	47
2	36
3	5
4	38
5	7
6	4
7	23
8	19
9	50

N=10

# Mediante exploración

6 -> hash(key) -> ?

0	95
1	47
2	36
3	5
4	38
5	7
6	4
7	23
8	19
9	50

N=10

# Factor carga

Es el grado de ocupación en una tabla Hash, sirve para conocer la probabilidad de colisiones, independientemente del algoritmo y su tratamiento de colisiones.

Factor carga = elementos / capacidad

Factor carga =  $(11 / 10) - 1$

Factor carga = 10%

# Factor carga

Si el factor de carga es mayor al 75%, el algoritmo tendrá serios problemas para tratar con las colisiones.

Factor carga = elementos / capacidad

Factor carga = (11 / 10) - 1

Factor carga = 10%

# Rehashing

Consiste en aumentar el tamaño de la tabla Hash para reducir el factor de carga y así disminuir la probabilidad de colisiones.

0	
1	
2	
3	
4	
5	
6	
7	
8	
9	

**N=10**



0	
1	
2	
3	
4	
5	
6	
7	
8	
9	
10	

**N=11**

# Aplicación de las tablas Hash en la informática

# Corrector ortográfico



## Corrección con un clic

Corrige cada error con un sólo clic.

ortografío

1

Hallado un posible error de ortografía

ortografía



añade "ortografío" a tu diccionario personal



# Diccionarios

```
1  # Create a dictionary
2
3  my_dict = {'Alex': 5,
4             'Ben' : 10,
5             'Carly': 12,
6             'Danielle': 7,
7             'Evan' : 6}
8  my_dict
```

```
{'Alex': 5, 'Ben': 10, 'Carly': 12, 'Danielle': 7, 'Evan': 6}
```

# Tablas de símbolos

## TABLA DE SÍMBOLOS

Operador	Significado
!	Negación
+	Suma
-	Resto
*	Multiplicación
/	División
%	Módulo
<	Menor
<=	Menor igual
>	Mayor
>=	Mayor igual
!=	Diferente
&&	Conjunción Lógica( Y)
	Disyunción Lógica( O)
==	Igualdad

**Shazam**





Universidad  
Rafael Landívar

Tradición Jesuita en Guatemala

# Tablas Hash

Colisiones y aplicaciones