



ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

CẤU TRÚC DỮ LIỆU VÀ GIẢI THUẬT BẢNG BẮM

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN, KHU PHỐ 6, PHƯỜNG LINH TRUNG, QUẬN THỦ ĐỨC, TP. HỒ CHÍ MINH

[T] 08 3725 2002 101 | [F] 08 3725 2148 | [W] www.uit.edu.vn | [E] info@uit.edu.vn



BẢNG BĂM (HASH TABLE)

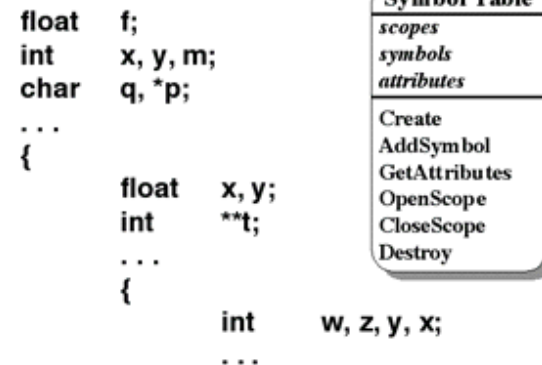
Nội dung

1. Đặt vấn đề
2. Bảng băm
3. Hàm băm
4. Giải quyết đụng độ
5. Bài tập áp dụng



1. ĐẶT VẤN ĐỀ

- Một số ứng dụng chỉ yêu cầu một bộ dữ liệu động, cần hỗ trợ 3 chức năng: INSERT, SEARCH, DELETE ...
 - Mỗi thành phần dữ liệu thường có các thuộc tính và dữ liệu (satellite data)
- **Từ điển**
- Ví dụ: Symbol table của một



<https://steemit.com/programming/@drifter1/writing-a-simple-compiler-on-my-own-symbol-table-basic-structure>



1. ĐẶT VẤN ĐỀ

- Bảng băm là một cấu trúc dữ liệu hiệu quả để hiện thực một từ điển.
- Theo giả định hợp lý, thời gian trung bình để tìm kiếm một yếu tố trong một bảng băm là **$O(1)$** .
 - Mảng: $O(1)$, chi phí lưu trữ lớn
 - Danh sách liên kết: $O(n)$
 - Cây nhị phân: $O(\log n)$

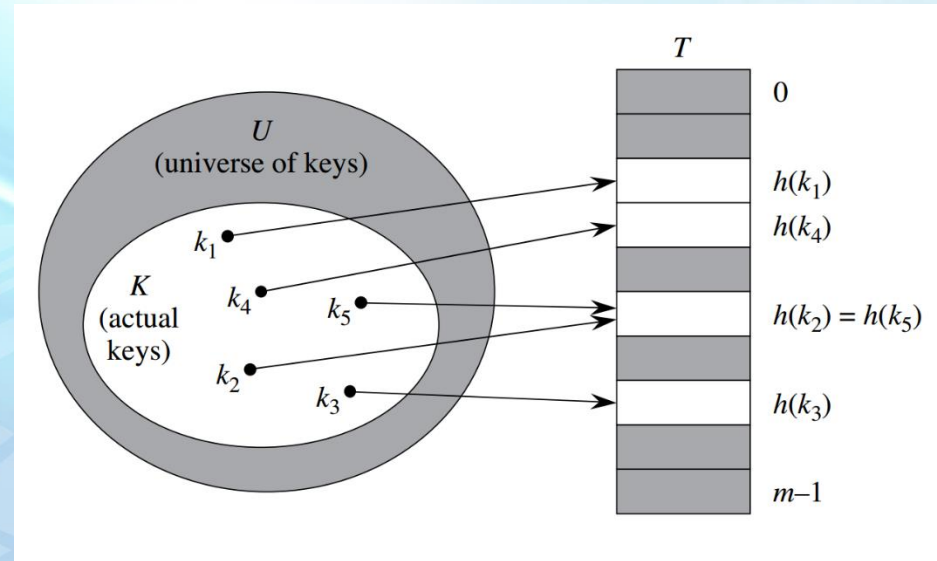


2. BẢNG BĂM

- Mỗi phần tử cần lưu trữ có khoá nằm trong tập khoá U .
- Hàm băm h ánh xạ phần tử trong cần lưu trữ vào vị trí $h(k)$ trong bảng băm có kích thước m :

$$h: U \rightarrow \{0, 1, \dots, m-1\}$$

- $m < |U|$, giảm chi phí lưu trữ
- Thời gian tìm kiếm $O(1)$

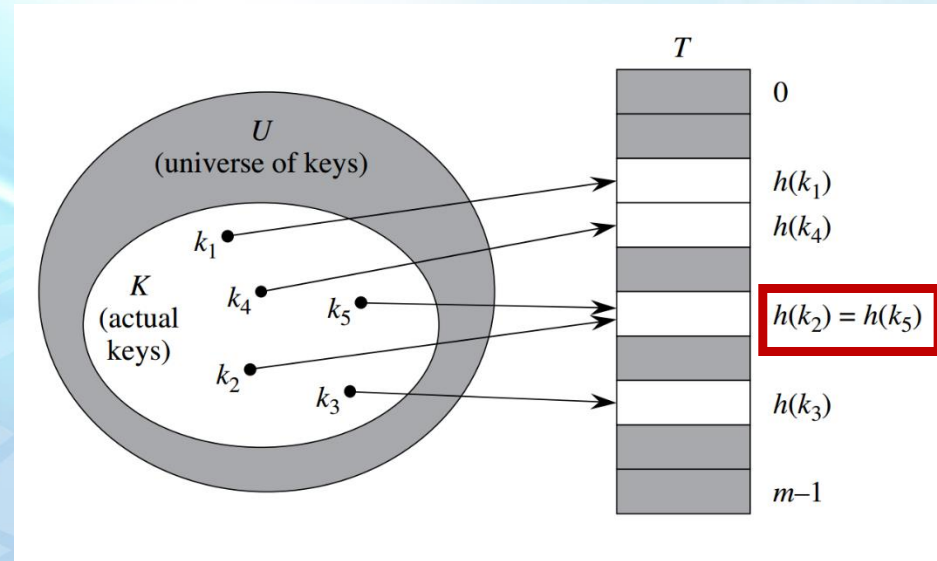


Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. 2009. *Introduction to Algorithms, Third Edition (3rd ed.)*. The MIT Press.



2. BẢNG BĂM

- Hai khoá có thể được lưu trữ cùng 1 vị trí trong bảng băm: Đụng độ (Collision)
- Giải quyết đụng độ:
 - Giảm đụng độ: Thiết kế hàm băm (Hash function)
 - Xử lý đụng độ: Chaining và Open Address



Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. 2009. *Introduction to Algorithms, Third Edition (3rd ed.)*. The MIT Press.



3. HÀM BẮM

KHÔNG CÓ
CÁCH NÀO ĐỂ
KIỂM TRA!!!!

- Tiêu chí hàm băm tốt: Khả năng mỗi khoá được băm vào vị trí bất kì trong m là tương đương, không phụ thuộc vào các khoá đã được băm trước đó.



- Chuyển các khoá về giá trị số tự nhiên
- Hàm băm sử dụng phương pháp chia
- Hàm băm sử dụng phương pháp nhân
- Hàm băm phổ quát



3. HÀM BĂM

Hàm băm tốt:

- ❖ Giảm thiểu đụng độ
- ❖ Tính toán dễ dàng và nhanh
- ❖ Phân phối các giá trị khóa đồng đều
- ❖ Sử dụng tất cả thông tin được cung cấp bởi khóa
- ❖ Có hệ số tải cao cho bộ khóa nhất định

Load Factor= số pt bảng băm/kích thước của bảng băm



3. HÀM BĂM

- Hàm băm sử dụng phương pháp chia:

$$h(k) = k \bmod m$$

k là khoá, m là kích thước của bảng.

- Chú ý: Chọn giá trị cho m
 - m không nên là lũy thừa của 2
 - m nên là 1 số nguyên tố không gần với một số là lũy thừa của 2

VD: Bảng băm có 4000 mục, chọn $m = 4093$



3. HÀM BĂM

VD: Bảng băm **11, 12, 13,14,15** vào mảng có size 5 sử dụng hàm băm

$$h(k) = k \bmod m$$



3. HÀM BĂM

VD: Bảng băm

Mia	M	77	i	105	a	97	279	4
Tim	T	84	i	105	m	109	298	1
Bea	B	66	e	101	a	97	264	0
Zoe	Z	90	o	111	e	101	302	5
Jan	J	74	a	97	n	110	281	6
Ada	A	65	d	100	a	97	262	9
Leo	L	76	e	101	o	111	288	2
Sam	S	83	a	97	m	109	289	3
Lou	L	76	o	111	u	117	304	7
Max	M	77	a	97	x	120	294	8
Ted	T	84	e	101	d	100	285	10

Bea	Tim	Leo	Sam	Mia	Zoe	Jan	Lou	Max	Ada	Ted
-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----

0 1 2 3 4 5 6 7 8 9 10



3. HÀM BĂM

VD: Bảng băm

Find Ada $262 \text{ Mod } 11 = 9$

MyData = Array(9)

Bea	Tim	Leo	Sam	Mia	Zoe	Jan	Lou	Max	Ada	Ted
0	1	2	3	4	5	6	7	8	9	10



3. HÀM BĂM

VD: Bảng băm **11, 12, 13, 14, 15** vào mảng có size 5 sử dụng hàm băm

$$h(k) = k \bmod m$$

VD2: Bảng băm **10, 11, 12, 13, 15** vào mảng có size 5 sử dụng hàm băm

$$h(k) = k \bmod m$$



3. HÀM BĂM

- Hàm băm sử dụng phương pháp nhân:

$$h(k) = \lfloor m * (k * A \bmod 1) \rfloor$$

k là khóa, m là kích thước bảng, A là hằng số: $0 < A < 1$

“ $k * A \bmod 1$ ”: phần thập phân của kA

- Chọn m và A :
 - Theo Knuth thì chọn A bằng giá trị sau: $A = (\sqrt{5} - 1)/2$
 - $m = 2^p$



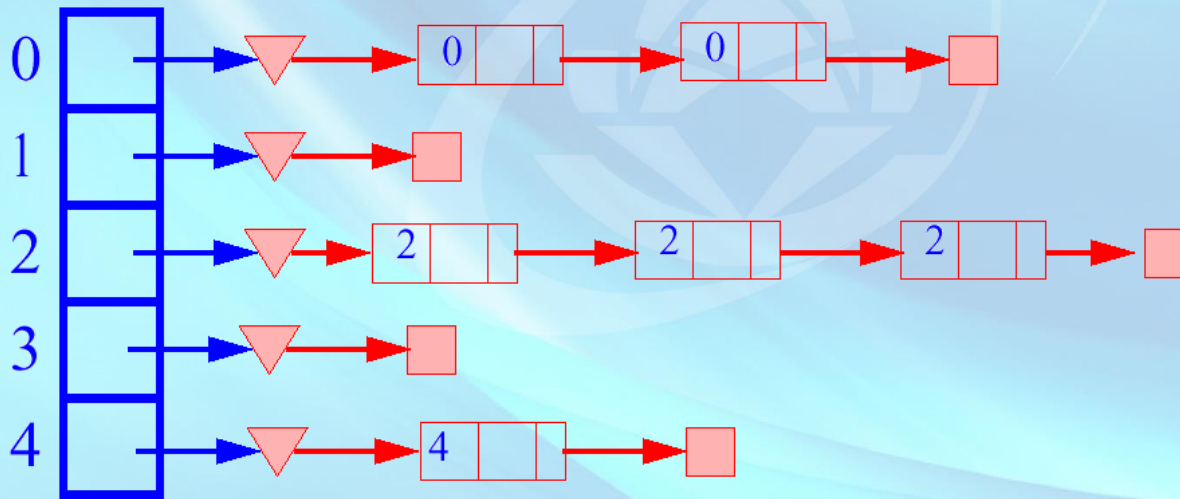
3. HÀM BẮM

- Hàm băm phổ quát: Việc chọn hàm băm không tốt có thể dẫn đến xác suất đụng độ cao.
- Giải pháp:
 - Lựa chọn hàm băm h ngẫu nhiên.
 - Khởi tạo một tập các hàm băm H phổ quát và từ đó h được chọn ngẫu nhiên.
- Cho H là một tập hợp hữu hạn các hàm băm: ánh xạ các khóa k từ tập khóa U vào miền giá trị $\{0,1,2,\dots, m-1\}$. Tập H là phổ quát nếu với mọi $f \in H$ và 2 khoá phân biệt k_1, k_2 ta có xác suất: $\Pr\{f(k_1) = f(k_2)\} \leq 1/m$



4. GIẢI QUYẾT ĐỤNG ĐỘ

- Phương pháp nối kết (Chaining)
 - Các nút bị băm cùng địa chỉ (các nút bị xung đột) được gom thành một danh sách liên kết.
 - Các nút bị xung đột tại địa chỉ i được nối kết trực tiếp với nhau qua danh sách liên kết i .

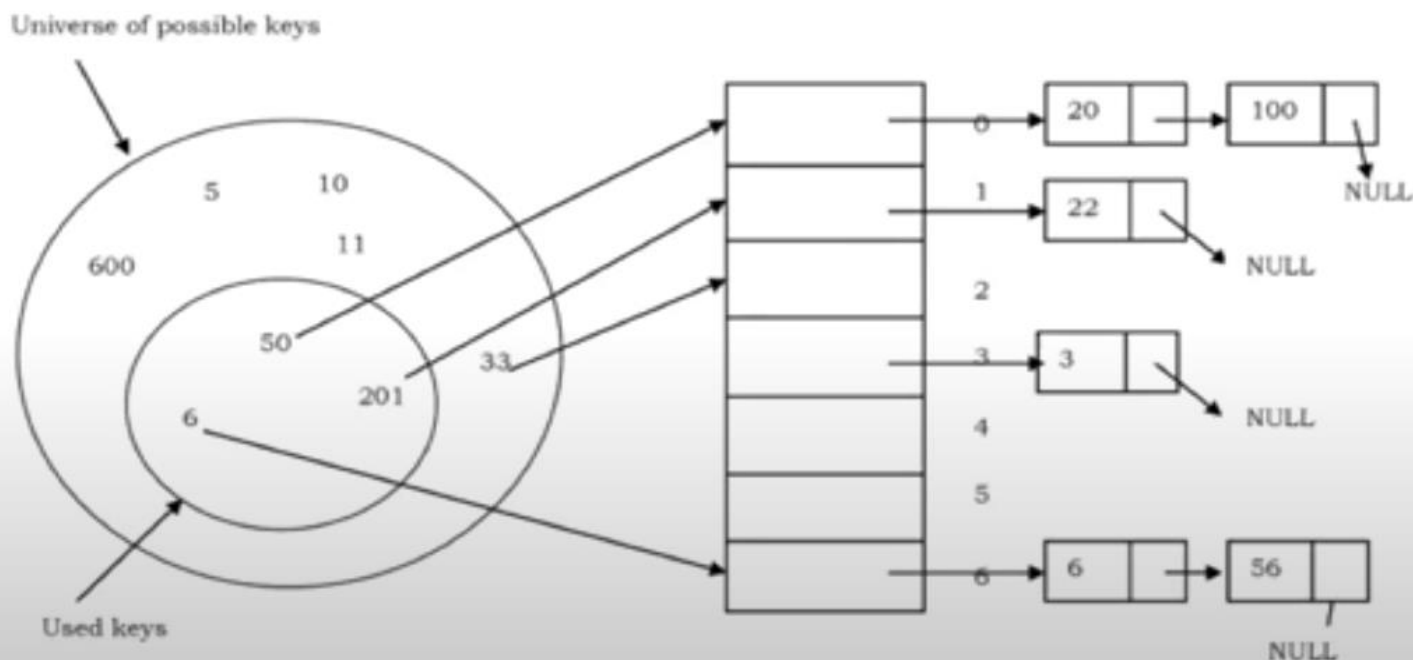




4. GIẢI QUYẾT ĐỤNG ĐỘ

- Ví dụ Phương pháp nối kết (Chaining)

Mỗi thành phần của trong bảng băm sẽ chứa một con trỏ trỏ tới danh sách liên kết





4. GIẢI QUYẾT ĐỤNG ĐỘ

- Phương pháp địa chỉ mở (Open Address)
- Ý tưởng: Nếu vị trí hiện tại đã bị khoá khác chiếm, ta sẽ thử tìm đến vị trí kế tiếp nào đó trong bảng.
- Dò tuyến tính (Linear probing):

$$h(k, i) = (h'(k) + i) \bmod m$$

$h'(k)$: hàm băm chính của bảng băm

$$i = 0, 1, \dots, m-1$$



4. GIẢI QUYẾT ĐỤNG ĐỘ

- Phương pháp địa chỉ mở (Open Address)

➤ Dò bậc hai (Quadratic probing):

$$h(k) = (h'(k) + c_1i + c_2i^2) \bmod m$$

$h'(k)$: hàm băm chính của bảng băm

c_1, c_2 là các hằng số ($c_1 = 0, c_2 = 1$)

$$i = 0, 1, \dots, m-1$$



4. GIẢI QUYẾT ĐỤNG ĐỘ

- Ví dụ Phương pháp địa chỉ mở (Open Address)

$h(\text{key}) = \text{key} \bmod 11$ **Băm dãy: 31, 19, 2, 13, 25, 24, 21, 9**

Insert keys

$$31 \bmod 11 = 9$$

$$19 \bmod 11 = 8$$

$$2 \bmod 11 = 2$$

$$13 \bmod 11 = 2 \rightarrow 2 + 1^2 = 3$$

$$25 \bmod 11 = 3 \rightarrow 3 + 1^2 = 4$$

$$24 \bmod 11 = 2 \rightarrow 2 + 1^2, 2 + 2^2 = 6$$

$$21 \bmod 11 = 10$$

$$9 \bmod 11 = 9 \rightarrow 9 + 1^2, 9 + 2^2 \bmod 11, 9 + 3^2 \bmod 11 = 7$$

0	
1	
2	2
3	13
4	25
5	5
6	24
7	9
8	19
9	31
10	21



4. GIẢI QUYẾT ĐỤNG ĐỘ

- Phương pháp địa chỉ mở (Open Address)

➤ Băm kép (Double Hashing):

$$h(k, i) = (h_1(k) + ih_2(k)) \bmod m$$

$h_1(k)$, $h_2(k)$: là 2 hàm băm bất kỳ

$$i = 0, 1, \dots, m-1$$



4. GIẢI QUYẾT ĐỤNG ĐỘ

- Ví dụ Phương pháp địa chỉ mở (Open Address)

Table size is 11 (0..10)

Hash Function: assume $h1(key) = key \bmod 11$ and $h2(key) = 7 - (key \bmod 7)$

Insert keys:

$58 \bmod 11 = 3$

$14 \bmod 11 = 3 \rightarrow 3 + 7 = 10$

$91 \bmod 11 = 3 \rightarrow 3 + 7, 3 + 2 * 7 \bmod 11 = 6$

$25 \bmod 11 = 3 \rightarrow 3 + 3, 3 + 2 * 3 = 9$

0	
1	
2	
3	58
4	25
5	
6	91
7	
8	
9	25
10	14



4. GIẢI QUYẾT ĐỤNG ĐỘ

- Phương pháp địa chỉ mở (Open Address)

LƯU Ý: Khi xóa một khóa từ vị trí i , không thể đơn giản đánh dấu vị trí đó là trống bằng cách lưu trữ NULL trong đó.

➔ Lưu trữ vị trí đó bằng 1 giá trị đặc biệt, vd như
DELETED



4. GIẢI QUYẾT ĐỤNG ĐỘ

- Phương pháp địa chỉ mở (Open Address):

HASH-DELETE (T, k)

```
1.      i = 0
2.      repeat
3.          j = h(k, i)
4.          if T[j] == k
5.              temp = T[j]
6.              T[j] = DELETED
7.              return temp
8.          else i = i + 1
9.      until T[j] == NULL or i == m
10.     return NULL
```



4. GIẢI QUYẾT ĐỤNG ĐỘ

- Phương pháp địa chỉ mở (Open Address)

HASH-INSERT (T, k)

```
1.      i = 0
2.      repeat
3.          j = h(k, i)
4.          if T[j] == NIL or T[j] == DELETED
5.              T[j] = k
6.              return j
7.          else i = i + 1
8.      until i == m
9.      error "hash table overflow"
```



4. GIẢI QUYẾT ĐỤNG ĐỘ

- Phương pháp địa chỉ mở (Open Address):

HASH-SEARCH (T, k)

```
1.      i = 0
2.      repeat
3.          j = h(k, i)
4.          if T[j] == k
5.              return j
6.          else i = i + 1
7.      until T[j] == NULL or i == m
8.      return NULL
```

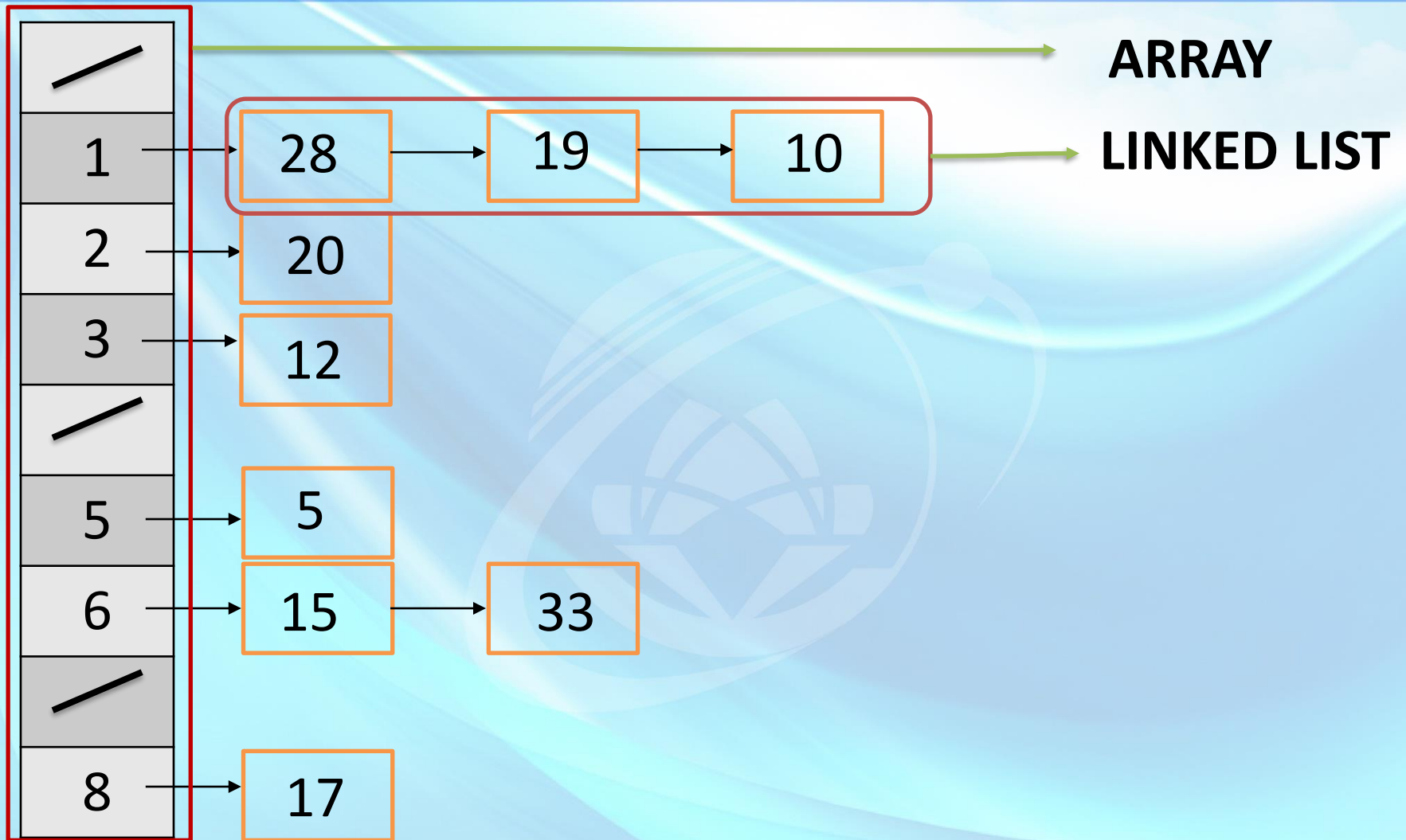


5. BÀI TẬP ÁP DỤNG

- Bài 1: Hãy mô tả các bước xảy ra khi chèn các khoá 5, 28, 19, 15, 20, 33, 12, 17, 10 vào một bảng băm được giải quyết đụng độ bằng phương pháp nối kết. Cho bảng băm có 9 ô và hàm băm là $h(k) = k \bmod 9$



5. BÀI TẬP ÁP DỤNG





5. BÀI TẬP ÁP DỤNG

- Bài 2: Xét một bảng băm có kích thước là $m = 1000$ và hàm băm là $h(k) = \lfloor m * (k * A \bmod 1) \rfloor$ với $A = (\sqrt{5} - 1)/2$. Hãy tính vị trí của các khoá 61, 62, 63, 65 khi được ánh xạ.



5. BÀI TẬP ÁP DỤNG

key	value
61	700
62	318
63	936
64	554
65	172



5. BÀI TẬP ÁP DỤNG

- Bài 3: Thêm các khoá 10, 22, 31, 4, 15, 28, 17, 88, 59 vào một bảng băm có kích thước $m = 11$ sử dụng địa chỉ mở với hàm băm $h'(k) = k$. Hãy minh hoạ kết quả khi thêm các khoá này vào bảng băm sử dụng phương pháp dò tuyến tính, dò bậc hai.



5. BÀI TẬP ÁP DỤNG

index	linear probing	quadratic probing	double hashing
0	22	22	22
1	88		
2		88	59
3		17	17
4	4	4	4
5	15		15
6	28	28	28
7	17	59	88
8	59	15	
9	31	31	31
10	10	10	10



BÀI TẬP

- Bài 1: Hãy mô tả các bước xảy ra khi chèn các khoá 5, 28, 19, 15, 20, 33, 12, 17, 10 vào một bảng băm được giải quyết đụng độ bằng phương pháp nối kết. Cho bảng băm có 9 ô và hàm băm là $h(k) = k \bmod 9$



BÀI TẬP

- Bài 2: Xét một bảng băm có kích thước là $m = 1000$ và hàm băm là $h(k) = \lfloor m * (k * A \bmod 1) \rfloor$ với $A = (\sqrt{5} - 1)/2$. Hãy tính vị trí của các khoá 61, 62, 63, 65 khi được ánh xạ.



BÀI TẬP

- Bài 3: Thêm các khoá 10, 22, 31, 4, 15, 28, 17, 88, 59 vào một bảng băm có kích thước $m = 11$ sử dụng địa chỉ mở với hàm băm $h(k) = k \bmod 11$.
Hãy minh họa kết quả khi thêm các khóa này vào bảng băm sử dụng phương pháp dò tuyến tính, dò bậc hai với $c_1 = 1$ và $c_2 = 3$, và băm kép với hàm băm phụ là $h'(k) = 1 + (k \bmod (m-1))$