



**Chương 6**  
**Tìm kiếm**  
(phần 2)


*Hiepnd@it-hut.edu.vn*

## Nội dung

- Bảng băm
- Một số phương pháp xây dựng hàm băm
- Độ phức tạp và khắc phục
- Bài tập

## Băm

- Các phương pháp tìm kiếm :
  - Tìm kiếm tuần tự:  $O(n)$
  - Tìm kiếm nhị phân  $O(\log n)$
- Phương pháp tổ chức dữ liệu để tìm kiếm nào tốt hơn ?

 **Bảng băm**

## Băm

- Ý tưởng bảng băm:
  - Dùng mảng lưu trữ các phần tử (bảng)
  - Mỗi phần tử sẽ được lưu tại một ô duy nhất trong bảng căn cứ vào giá trị khóa của nó,
  - Khi tìm kiếm thì căn cứ vào khóa cần tìm ta tìm đến ô tương ứng
  - Nếu ô đó có chứa phần tử thì tìm thấy, ngược lại là không tìm thấy

## Băm

- VD. Sinhvien(1222, 'Nguyễn văn A', 'Hà Nội')  
Sinhvien(1101, 'Trần văn C', 'Thái Bình')  
Sinhvien(0097, 'Nguyễn văn A', 'Hà Nội')  
Sinhvien(1331, 'Trần văn C', 'Thái Bình')  
Sinhvien(1345, 'Nguyễn văn A', 'Hà Nội')
- Sử dụng bảng kích thước 100 để lưu các phần tử

Chỉ số mảng	1	Sinhvien(1101, 'Trần văn C', 'Thái Bình')
	...	
	22	Sinhvien(1222, 'Nguyễn văn A', 'Hà Nội')
	...	
	31	Sinhvien(1331, 'Trần văn C', 'Thái Bình')
	...	
	35	Sinhvien(1345, 'Nguyễn văn A', 'Hà Nội')
	...	
	97	Sinhvien(0097, 'Nguyễn văn A', 'Hà Nội')

## Băm

- Tốt hơn tìm kiếm tuần tự khi mà dữ liệu được lưu trữ không có thứ tự.
- Không dựa trên so sánh giá trị các khóa của bản ghi mà dựa trên chính bản thân giá trị các khóa.
- Từ giá trị khóa ta tính ra một địa chỉ (địa chỉ tương đối) theo một quy tắc nào đó. Địa chỉ này sẽ dùng để lưu trữ bản ghi và cũng để tìm kiếm bản ghi đó.

hash function: key → an index  
Địa chỉ lưu trữ

- Hàm băm hoàn hảo: mỗi khóa ứng với một giá trị nguyên duy nhất
- Thời gian tìm kiếm  $O(1)$  ???

## Băm

- Hàm băm:
  - Tính toán dễ và nhanh
  - Phân phối đều các khóa

Một số phương pháp thông dụng:

- Cắt bỏ (Truncation)**: dùng một phần của khóa làm chỉ số  
VD: khóa có 8 chữ số và bảng băm kích thước 1000 lấy chữ số thứ 4, 7 và 8 làm chỉ số  
21296876 → 976  
Nhanh nhưng phân bố khóa không đều!

## Băm

- Gấp (folding)**: chia khóa thành nhiều phần sau đó kết hợp các phần lại (thường dùng cộng hoặc nhân)

VD. 21296876 chia thành 3 phần 212, 968 và 76  
kết hợp  $212+968+76 = 1256$ , cắt bỏ được 256

Giá trị các thành phần trong khóa đều ảnh hưởng tới chỉ số. Cho phân phối tốt hơn phương pháp cắt bỏ

## Băm

- **Phương pháp chia module:** lấy số dư của phép chia giá trị khóa cho kích thước của bảng băm để làm địa chỉ

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

- Thường chọn m là số nguyên tố nhỏ hơn, gần với kích thước bảng băm.
  - Bảng băm kích thước 1000 thì chọn m=997

Giá trị khóa	địa chỉ
5402	417
367	367
1246	249
2983	989

## Băm

- **Phương pháp nhân:** giá trị khóa được nhân với chính nó, sau đó lấy một phần kết quả để làm địa chỉ băm

Giá trị khóa k	k*k	địa chỉ
5402	29181604	181
367	00134689	134
1246	01552516	552
2983	08898289	898

## Băm

- **Đụng độ:** hai khóa khác nhau có cùng giá trị chỉ số

VD: hai khóa 21296876, 11391176



Trong hàm băm dùng phương pháp cắt bỏ

- Giải pháp xử lý đụng độ:
  - Phương pháp đánh địa chỉ mở (Open Addressing)
  - Phương pháp xích ngăn cách (Separate Chaining)

## Phương pháp đánh địa chỉ mở

### Phương pháp đánh địa chỉ mở - Open Addressing

- **Dò tuyến tính (Linear Probing):** tại vị trí đụng độ, thực hiện tìm kiếm tuần tự để tìm ra khóa (khi tìm kiếm) hoặc vị trí trống (khi thêm mới)

VD: hàm băm

$$i = k \% 10$$

Các khóa {89, 18, 49, 58, 69} được thêm lần lượt vào bảng băm ban đầu rỗng

## Phương pháp đánh địa chỉ mở

stt	Ban đầu	Thêm 89	Thêm 18	Thêm 49	Thêm 58	Thêm 69
0				49	49 (*)	49 (*)
1					58	58 (*)
2						69
3						
4						
5						
6						
7						
8			18	18	18 (*)	18
9		89	89	89 (*)	89 (*)	89 (*)

## Phương pháp đánh địa chỉ mở

### ■ Dò tuyến tính

- Xu hướng tạo thành các cụm khi bảng bắt đầu gần đầy nửa
- Vị trí lưu trữ của khóa trong bảng và giá trị chỉ số ngày càng cách nhau xa, chi phí thực hiện tìm kiếm tuần tự ngày càng lớn

### ■ **Khắc phục:** sử dụng các phương pháp lựa chọn vị trí phức tạp khi xảy ra đụng độ

VD. Phương pháp băm lại (rehashing) sử dụng hàm băm thứ 2 để tạo chỉ số khi xảy ra đụng độ, nếu lại đụng độ thì sử dụng hàm băm thứ 3 ....

## Phương pháp đánh địa chỉ mở

- **Dò bậc hai, dò toàn phương** (Quadratic Probing): nếu xảy ra đụng độ tại địa chỉ  $h$ , thì ta sẽ tìm kiếm tiếp theo tại các địa chỉ  $h+1, h+4, h+9, \dots$ ; là các địa chỉ  $h+i^2$

VD: hàm băm

$$i = k \% 10$$

Các khóa {89, 18, 49, 58, 69} được thêm lần lượt vào bảng băm ban đầu rỗng

## Phương pháp đánh địa chỉ mở

stt	Ban đầu	Thêm 89	Thêm 18	Thêm 49	Thêm 58	Thêm 69
0				49	49	49 (*)
1						
2					58	58
3						69
4						
5						
6						
7						
8			18	18	18 (*)	18
9		89	89	89 (*)	89 (*)	89 (*)

## Phương pháp đánh địa chỉ mở

### ■ Dò toàn phương:

- Giảm được sự phân nhóm
- Không phải tất cả các vị trí trong bảng đều được dò

VD. Khi kích thước bảng là mũ của 2 thì 1/6 vị trí được dò, là số nguyên tố thì 1/2 được dò

## Phương pháp đánh địa chỉ mở

- **Dò theo khóa:** trong trường hợp đựng độ tại h thì dò tiếp tại vị trí cách vị trí đó khoảng cách bằng giá trị phần tử tiên trong khóa (là số hoặc là mã ASCII).

- Ví dụ: nếu khóa **2918160** xảy ra đựng độ thì dò tại ô tiếp theo cách ô đựng độ 2 vị trí

## Phương pháp đánh địa chỉ mở

- **Dò ngẫu nhiên** (Random Probing): sử dụng cách sinh số giả "ngẫu nhiên" để tạo ra vị trí dò tiếp. Cách sinh này phải là duy nhất với 1 giá trị khóa.

- Ví dụ: khóa 123245 thì sinh ra số ngẫu nhiên duy nhất là 5

## Phương pháp đánh địa chỉ mở

- **Chú ý:** không thể xóa phần tử trong bảng băm sử dụng phương pháp đánh địa chỉ mở theo cách thông thường!  
→ đánh dấu là xóa, nhưng vẫn được xét đến khi dò!



## Giải pháp xử lý đụng độ

### Phương pháp xích ngăn cách (Separate Chaining)

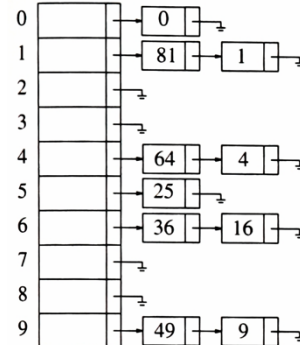
■ Dùng mảng của danh sách móc nối

■ Ưu điểm:

- không bị giới hạn số phần tử,
- Dễ thêm và xóa
- xử lý đụng độ tốt

■ Nhược điểm :

- Phải lưu nhiều con trỏ NULL
- Thực hiện tìm kiếm tuyến tính nên thời gian tìm kiếm chậm



## Băm

■ Hàm băm tương ứng của số có 3 chữ số (  $a_1 a_2 a_3$  ) là  $\text{mod}(a_1 + a_2 + a_3, 5)$ . Với các tổ hợp số dưới đây, giá trị băm bị xung đột là trường hợp nào?

- A. 881 và 509
- B. 913 và 426
- C. 731 và 429
- D. 102 và 297
- E. 677 và 388

$\text{mod}$ : chia lấy phần dư       $\text{mod}(5,3) = 2$

## Băm

**Bài 2.** Cho bảng băm với kích thước 13, chỉ số các phần tử từ 0 đến 12, và dãy khóa 10, 100, 32, 45, 58, 126, 3, 29, 200, 400, 0

- a) Sử dụng hàm băm  $i = k \% 13$ , vẽ các bước khi thêm các khóa vào bảng sử dụng phương pháp xử lý đụng độ là dò tuyến tính và dò bậc hai.
- b) Sử dụng hàm băm là tổng của các chữ số trong khóa chia lấy dư cho 13, vẽ lại bảng băm với hai phương pháp xử lý đụng độ như phần a
- c) Tìm một hàm băm mà không xảy ra đụng độ cho dãy khóa trên