

# Cấu trúc dữ liệu và giải thuật

**TS. Phạm Tuấn Minh**

Khoa Công nghệ Thông tin, Đại học Phenikaa

[minh.phamtuan@phenikaa-uni.edu.vn](mailto:minh.phamtuan@phenikaa-uni.edu.vn)

<https://sites.google.com/site/phamtuanminh/>

## Chương 3: Cây và bảng băm

- ❑ Các khái niệm cây
- ❑ Cây nhị phân tìm kiếm
- ❑ Cây AVL
- ❑ Bảng băm

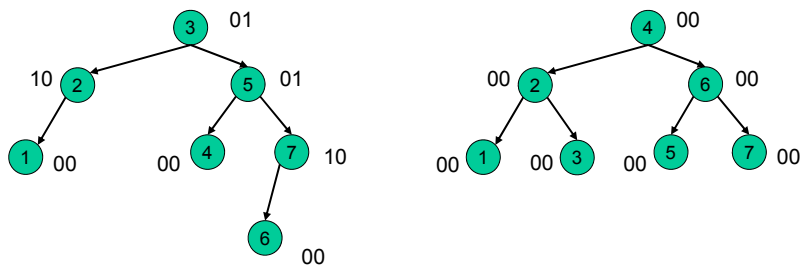
## Cây AVL

- ❑ **Khái niệm và đặc điểm cây AVL**
- ❑ Bổ sung trên cây AVL

1-3

## Khái niệm cây AVL

- ❑ Cây nhị phân tìm kiếm được gọi là cân đối AVL nếu như đối với mọi nút của nó chiều cao của hai cây con tương ứng chỉ chênh nhau một đơn vị



- ❑ Hệ số cân đối (balance factor):
  - 00 ứng với cây con trái và cây con phải có chiều cao bằng (cân bằng)
  - 01 ứng với cây con phải có chiều cao lớn hơn 1 (lệch phải)
  - 10 ứng với cây con trái có chiều cao lớn hơn 1 (lệch trái)

1-4

## Đặc điểm của cây AVL

- ❑ Tính cân đối của cây AVL có giảm nhẹ hơn so với tính cân đối của cây nhị phân hoàn chỉnh hoặc gần đầy: Cây nhị phân hoàn chỉnh hoặc gần đầy bao giờ cũng là cây cân đối AVL nhưng cây AVL thì chưa chắc đã là cây nhị phân hoàn chỉnh hoặc gần đầy
- ❑ Adelson - Velski và Landis đã chứng minh được rằng chiều cao của cây cân đối AVL chỉ vượt hơn khoảng 45% so với chiều cao của cây nhị phân hoàn chỉnh. Cụ thể là nếu gọi chiều cao của cây cân đối AVL có  $n$  nút là  $h_{AVL}(n)$  thì  $\log_2(n+1) < h_{AVL}(n) < 1,4404 \log_2(n+2) - 0,328$
- ❑ Nếu cây nhị phân tìm kiếm mà luôn luôn có dạng cân đối AVL, thì chi phí cho tìm kiếm đối với nó, ngay trong trường hợp xấu nhất vẫn là  $O(\log_2 n)$

1-5

## Cây AVL

- ❑ Khái niệm và đặc điểm cây AVL
- ❑ **Bổ sung trên cây AVL**

1-6

## Bổ sung trên cây AVL

- ❑ Thêm một trường BIT để ghi nhận hệ số cân đối của nút (chỉ cần 2 bit)
- ❑ Việc đi theo đường tìm kiếm trên cây để thấy được khoá mới chưa có sẵn trên cây và biết được "chỗ" để bổ sung nó vào, có một điều khác so với tìm kiếm trên cây nhị phân thông thường là đường đi này phải được ghi nhận lại để phục vụ cho việc xem xét và chỉnh lý hệ số cân đối của nút trên đường đi đó, bị tác động bởi phép bổ sung.

1-7

## Bổ sung trên cây AVL

- ❑ Giả sử phép bổ sung được thực hiện vào phía trái (phía phải thì cũng tương tự), sau khi nút mới được bổ sung, có ba tình huống có thể xảy ra với các nút tiền bối của nó:
  - Tình huống 1: Cây con phải đã cao hơn 1 (lệch phải) sau phép bổ sung chiều cao hai cây con bằng nhau
  - Tình huống 2: Chiều cao hai cây con vốn đã bằng nhau, sau phép bổ sung cây con trái cao hơn 1 (lệch trái)
  - Tình huống 3: Cây con trái đã cao hơn 1 (lệch trái), sau phép bổ sung nó cao hơn 2: tính "cân đối AVL" bị phá vỡ

1-8

## Bổ sung trên cây AVL: Tình huống 1

- Tình huống 1: Cây con phải đã cao hơn 1 (lệch phải) sau phép bổ sung chiều cao hai cây con bằng nhau
  - Chỉ cần chỉnh lý các hệ số cân đối ở nút đang xét

1-9

## Bổ sung trên cây AVL: Tình huống 2

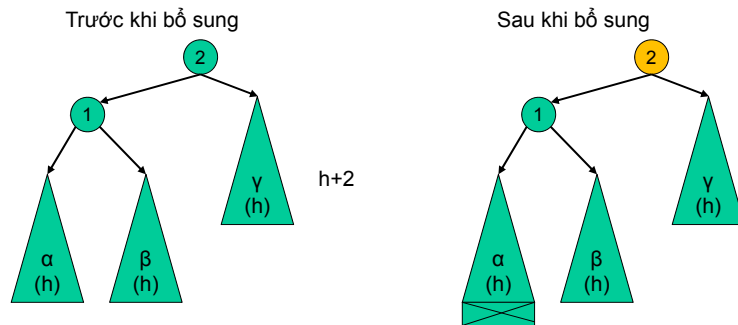
- Tình huống 2: Chiều cao hai cây con vốn đã bằng nhau, sau phép bổ sung cây con trái cao hơn 1 (lệch trái)
  - Chiều cao của cây có gốc là nút đang xét bị thay đổi, nên không những phải chỉnh lý hệ số cân đối ở nút đang xét mà còn phải chỉnh lý hệ số cân đối ở các nút tiền bối của nó.
  - Dọc trên đường đi đã ghi nhận khi tìm kiếm, cũng phải xem xét để biết có xảy ra tình huống nào trong ba tình huống đã nêu đối với nút đó không và có biện pháp xử lý thích hợp. Như vậy có khi cần phải lần ngược lại tới tận gốc cây.

1-10

## Bổ sung trên cây AVL: Tình huống 3

- Tình huống 3: Cây con trái đã cao hơn 1 (lệch trái), sau phép bổ sung nó cao hơn 2: tính "cân đối AVL" bị phá vỡ
  - Đòi hỏi phải sửa lại cây con mà nút đang xét là nút gốc (ta sẽ gọi là "nút bất thường" critical node), để nó cân đối lại.

Trường hợp 1 (LL): Nút mới bổ sung làm tăng chiều cao **cây con trái của nút con trái** nút bất thường.



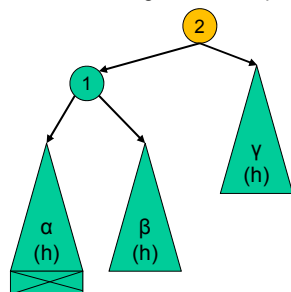
Nút mới bổ sung làm tăng chiều cao của cây  $\alpha$ , cây con trái của nút 1

1-11

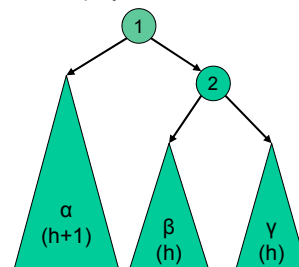
## Bổ sung trên cây AVL: Tình huống 3

- Trường hợp 1 (LL): Nút mới bổ sung làm tăng chiều cao **cây con trái của nút con trái** nút bất thường.
  - Để tái cân đối ta phải thực hiện một phép quay từ trái sang phải để đưa nút (1) lên vị trí gốc cây con, nút (2) sẽ trở thành con phải của nó, và p được gắn vào thành con trái của (2) (Gọi là phép quay đơn)

Sau khi bổ sung, trước khi quay



Sau khi quay

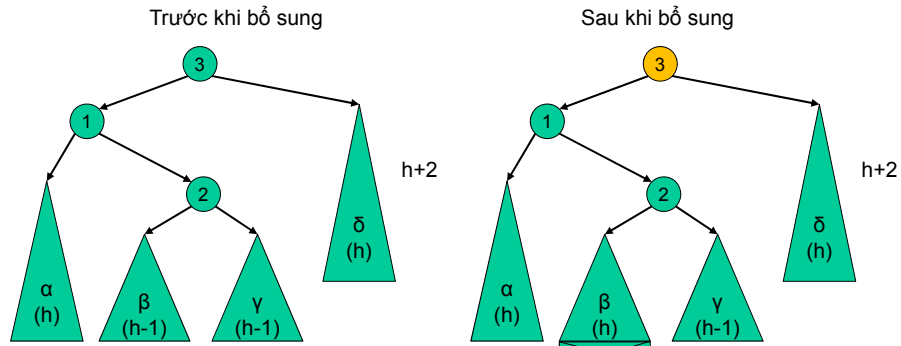


1-12

## Bổ sung trên cây AVL: Tình huống 3

- Tình huống 3: Cây con trái đã cao hơn 1 (lệch trái), sau phép bổ sung nó cao hơn 2: tính "cân đối AVL" bị phá vỡ
  - Đòi hỏi phải sửa lại cây con mà nút đang xét là nút gốc (ta sẽ gọi là "nút bất thường" critical node), để nó cân đối lại.

Trường hợp 2 (LR): Nút mới bổ sung làm tăng chiều cao **cây con phải của nút con trái** nút bất thường.



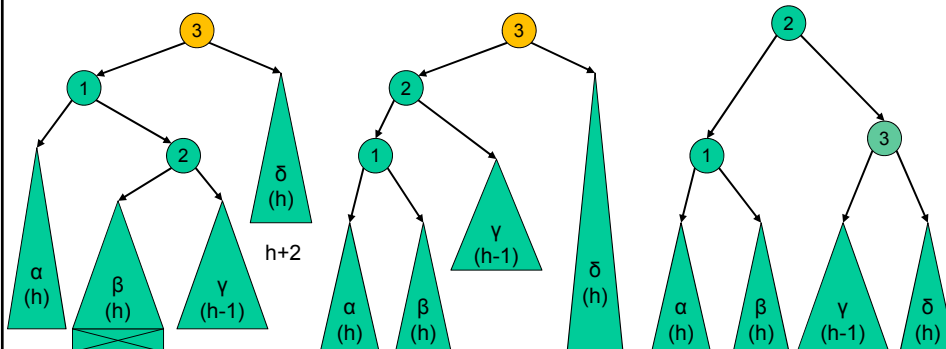
Nút mới bổ sung làm tăng chiều cao của cây  $\beta$ , cây con phải của nút 1

1-13

## Bổ sung trên cây AVL: Tình huống 3

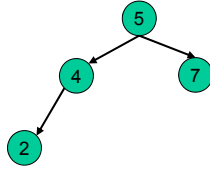
- Trường hợp 2 (LR): Nút mới bổ sung làm tăng chiều cao cây con phải của nút con trái nút bất thường.
  - Để tái cân đối ta phải thực hiện một phép quay kép (double rotation), đó là việc phối hợp hai phép quay đơn: **quay trái đối với cây con trái** ((1), (2)) và **quay phải đối với cây** ((3), (2))

Sau khi bổ sung, trước khi quay      Sau phép quay đơn thứ nhất      Sau phép quay đơn thứ hai

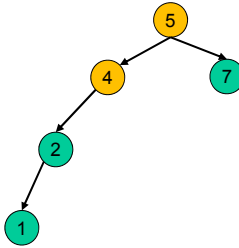


1-14

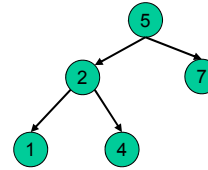
## Ví dụ



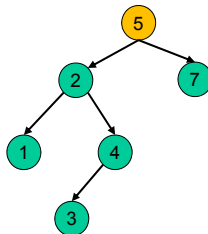
Cây ban đầu



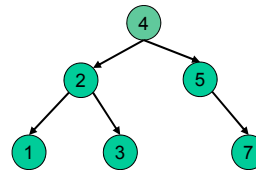
Bổ sung thêm nút (1): mất cân đối (trường hợp 1)



Tái cân đối bằng mất cân đối bằng phép quay đơn



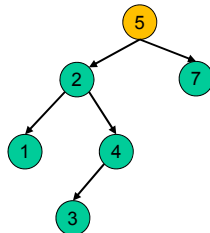
Bổ sung thêm nút (3): mất cân đối (trường hợp 2)



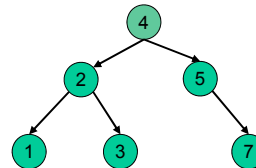
Tái cân đối bằng phép quay kép

1-15

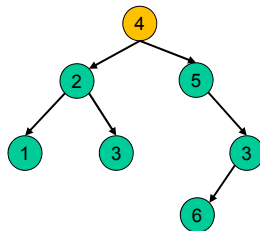
## Ví dụ



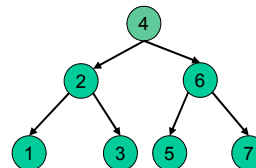
Bổ sung thêm nút (3): mất cân đối (trường hợp 2)



Tái cân đối bằng phép quay kép



Bổ sung thêm nút (6): mất cân đối (trường hợp 2)



Tái cân đối bằng phép quay kép

1-16



## Bổ sung trên cây AVL

### □ Nhận xét:

- Sau khi thực hiện phép quay để tái cân đối cây con mà nút gốc là nút bất thường thì chiều cao của cây con đó vẫn giữ nguyên như trước lúc bổ sung, nghĩa là phép quay không làm ảnh hưởng gì tới chiều cao các cây có liên quan tới cây con này.
- Ngoài ra tính chất của cây nhị phân tìm kiếm cũng luôn luôn được đảm bảo
- Phép loại bỏ cũng được thực hiện tương tự với chi phí không khác gì lắm so với phép bổ sung. Phép loại bỏ có thể cũng gây ra mất cân đối và phải tái cân đối bằng các phép quay như đã làm khi bổ sung

1-17

## Cấu trúc dữ liệu và giải thuật

- Nội dung bài giảng được biên soạn bởi TS. Phạm Tuấn Minh.

1-18