# Đồ án cuối năm cây AVL nhóm 15

## 1/Định nghĩa cây AVL

### +Lí thuyết về cây AVL:

Là cây tìm kiếm nhị phân tự cân bằng và cấu trúc dữ liệu đầu tiên có khả năng này do luôn đảm bảo điều kiện độ cao của các cây con bên trái và cây con bên phải và bảo đảm rằng hiệu số giữa chúng là không lớn hơn 1. **Hiệu số** này được gọi là **Balance Factor (Nhân tố cân bằng)**.

. Về những phần tử của cây AVL bao gồm:

+Node gốc.(Thường phần tử nhập vào đầu tiên trong cây

+Các node phần tử(liên kết với nhau bởi các con trỏ trái và phải với cấu trúc theo dạng xếp dần xuống dưới,1 node có tối đa 2 cây trái và phải->Cấu trúc dạng cây) theo qui tắc:

+Cây con trái(với những phần tử thêm vào sẽ được so sánh với node gốc->nếu giá trị node thêm vào nhỏ hơn node gốc sẽ được xếp vào cây con trái).

+Cây côn phải(với những phần tử thêm vào sẽ được so sánh với node gốc->nếu giá trị node thêm vào lớn hơn node gốc sẽ được xếp vào cây con phải).

\*Note:Trong cây AVL không tồn tại phần tử bằng nhau

### +Mục đích sử dụng:

Giúp việc tìm kiếm phần tử dễ dàng hơn so với việc sử dụng mảng 1 chiều và danh sách liên kết đơn với thời gian làm việc là **O(log n)** với mội trường hợp dù là xấu nhất do việc có thể giảm bớt số phần tử khi duyệt cây(việc duyệt chỉ xảy ra với cây con trái hoặc phải->loại bỏ bớt trường hợp không cần thiết phải duyệt ngày từ đầu) và luôn đảm bảo tính đúng đắn, ổn định do rang buộc và tính tự cân bằng của cây.

## 2/Xây dựng thuật toán tạo cây(phần này xử lí trên code):

## 3/ Xây dựng thuật toán thao tác trên cây:

Khi chèn thêm hoặc xóa đi 1 phần tử v cho nút u cây AVL cân bằng sẽ dẫn sinh các trường hợp:Nếu với mọi đỉnh u của T ta có **balance**(u)= 0 thì T được gọi là cây cân bằng hoàn toàn; do v chỉ có thể được chèn vào đúng một trong hai cây con của **u** nên nhiều nhất là **v** có thể làm tăng chiều cao của một trong hai cây con đó Nếu balance(T) > 0, nghĩa là cây con trái cao hơn cây con phải T được gọi là cây lệch trái; Nếu balance(T)< 0, nghĩa là cây con phải cao hơn cây con trái T được gọi là cây lệch phải.Cụ thể:

-Trước khi chèn cây con gốc u lệch trái và v làm tăng chiều cao của cây con trái.

+Sau khi chèn cây con trái lệch trái (Case LL):cách giải quyết sử dụng kĩ thuật quay phải:



+Sau khi chèn cây con trái lệch phải (Case LR):cách giải quyết sử dụng phép quay trái-phải:

|  |  |
| --- | --- |
| Trạng thái | Hành động |
|  | Một nút đã được chèn vào cây con bên phải của cây con bên trái.Điều này làm nút C trở nên không cân bằng |
|  | Đầu tiên, thực hiện phép quay trái trên cây con bên trái của **C**. Điều này làm cho **A** trở thành cây con bên trái của **B**. |
|  | Bây giờ nút **C** vẫn không cân bằng, đó là do xuất hiện cây con bên trái của cây con bên trái. |
|  | Bây giờ chúng ta sẽ thực hiện kỹ thuật quay phải để làm **B** trở thành nút gốc mới của cây này. Nút **C** bây giờ trở thành cây con bên phải của chính cây con bên trái của nó. |
|  | Cây trở lại trạng thái cân bằng |

-Trước khi chèn cây con gốc u lệch phải và v làm tăng chiều cao của cây con phải.

+Sau khi chèn cây con phải lệch phải (Case RR):cách giải quyết sử dụng kĩ thuật quay trái:



+Sau khi chèn cây con phải lệch trái. (Case RL): cách giải quyết sử dụng kĩ thuật quay phải-trái:

|  |  |
| --- | --- |
| Trạng thái | Hành động |
|  | Một nút đã được chèn vào trong cây con bên trái của cây con bên phải. Điều này làm nút **A** trở nên không cân bằng |
|  | Đầu tiên, chúng ta thực hiện kỹ thuật quay phải với nút **C**, làm cho **C** trở thành cây con bên phải của chính cây con bên trái **B**. Bây giờ, nút **B** trở thành cây con bên phải của nút **A**. |
|  | Bây giờ nút **A** vẫn không cân bằng bởi vì xuất hiện cây con bên phải của cây con bên phải của nó. Do đó cần phải thực hiện một kỹ thuật quay trái. |
|  | Một kỹ thuật quay trái được thực hiện làm cho **B** trở thành nút gốc mới của cây con. Nút **A** trở thành cây con bên trái của cây con **B** bên phải của nó. |
|  | Một kỹ thuật quay trái được thực hiện làm cho **B** trở thành nút gốc mới của cây con. Nút **A** trở thành cây con bên trái của cây con **B** bên phải của nó. |

## 3/Thuật toán xuất cây(phần này đi theo code):