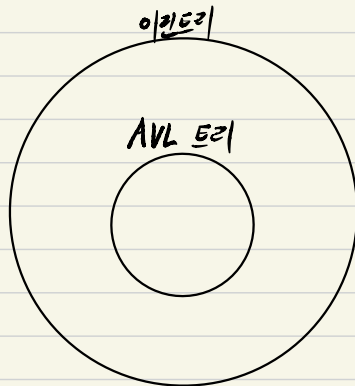




## AVL 트리

AVL 트리:  $\text{abs}(\text{왼쪽 서브 트리 높이} - \text{오른쪽 서브 트리 높이}) \leq 1$  인 **이진트리**



AVL 트리는 균형을 보장한다.  $\rightarrow$  회전, 좌회전, 우회전이 모두  $O(\log n)$

## 균형인수

균형인수 (balance factor):  $\text{왼쪽 서브 트리 높이} - \text{오른쪽 서브 트리 높이}$

모든 노드의  $\text{abs}(\text{균형인수}) \leq 1$  이어야 AVL 트리이다.

## 균형이 깨지는 경우

$\rightarrow$  노드의 삽입/삭제 시, 균형이 깨질 수 있다.

다시 균형을 맞추려면?  $\rightarrow$  균형인수가  $\pm 2$ 가 된 가장 가까운 조상 노드를 재배치

## 4가지 경우

삽입 연산으로 인해, 균형인수가  $\pm 2$ 가 된 가장 가까운 조상 노드:  $x$

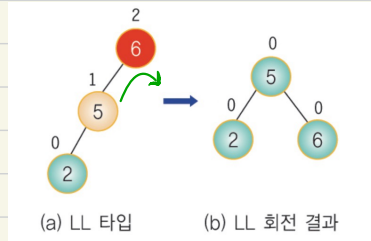
LL타입:  $x$ 의 왼쪽 자식노드의 왼쪽 자식노드가 추가됨

LR타입:  $x$ 의 왼쪽 자식노드의 오른쪽 자식노드가 추가됨

RR타입:  $x$ 의 오른쪽 자식노드의 오른쪽 자식노드가 추가됨

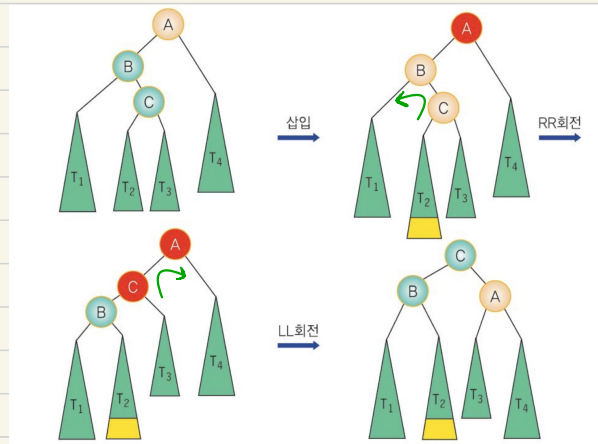
RL타입:  $x$ 의 오른쪽 자식노드의 왼쪽 자식노드가 추가됨

## 회전방법



LL 타입: X와 그 자식노드를 오른쪽으로 회전

return: 회전 후 root



LR 타입: X의 자식노드와 그것의 자식노드를 RR 회전 (왼쪽으로 회전)  
 즉, X와 X의 자식노드를 LL 회전 (오른쪽으로 회전)