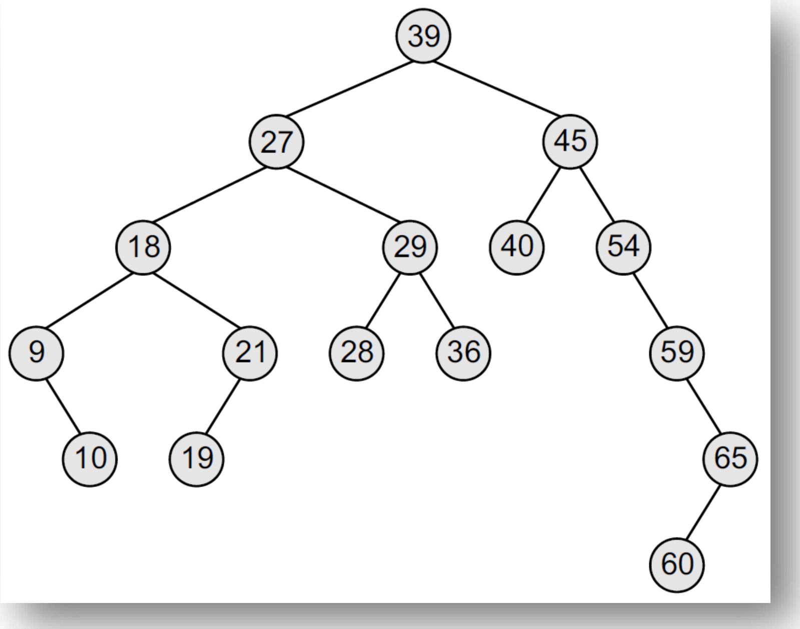
AVL Trees

Kuan-Yu Chen (陳冠宇)

2020/10/21 @ TR-212, NTUST

Review

- Binary search tree is also known as **ordered binary tree**
  - All the nodes in the **left sub-tree** have a value **less** than that of the root node
  - All the nodes in the **right sub-tree** have a value either **equal to or greater** than the root node



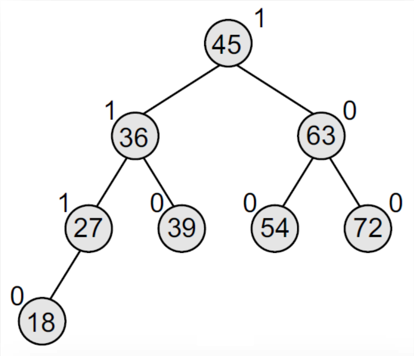
AVL Trees.

- AVL tree is a self-balancing binary search tree
  - AVL tree is designed by G.M. Adelson-Velsky and E.M. Landis in 1962
  - The heights of the two sub-trees of a node may differ by at most one

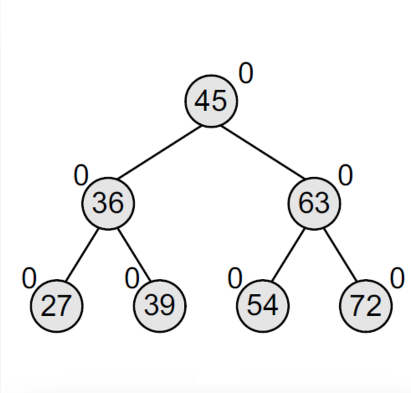
- The structure of an AVL stores an additional variable called the **Balance Factor**
  - Every node has a balance factor
  - The balance factor of a node is calculated by subtracting the height of its right sub-tree from the height of its left sub-tree
  - Every node has a balance factor of –1, 0, or 1

AVL Trees..

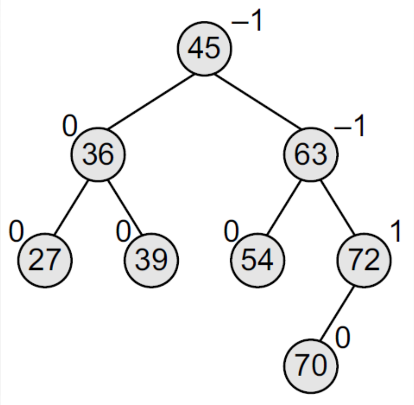
- If the balance factor of root is 1, then it means that the left sub-tree of the tree is one level higher than that of the right sub-tree
  - Left-heavy tree



- If the balance factor of root is 0, then it means that the height of the left sub-tree is equal to the height of the right sub-tree
  - Balance tree



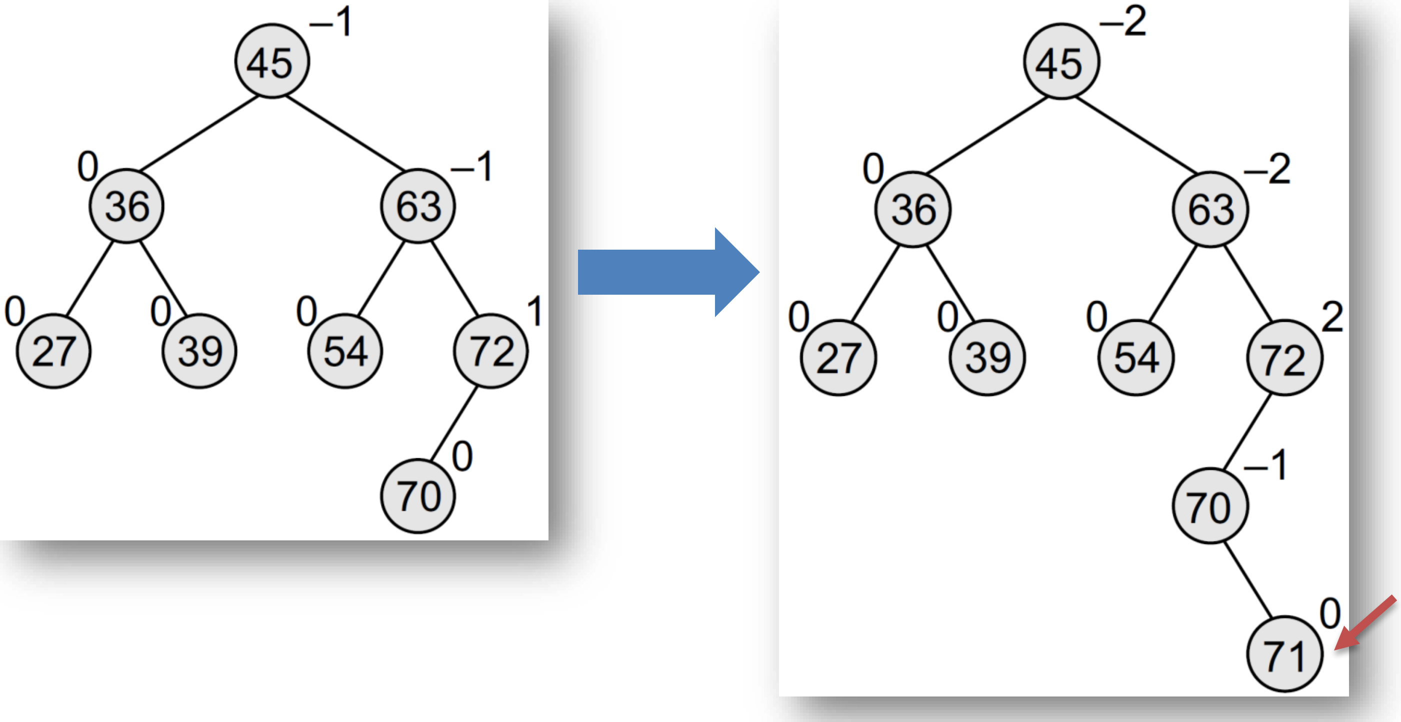
- If the balance factor of root is –1, then it means that the left sub-tree of the tree is one level lower than that of the right sub-tree
  - Right-heavy tree



Insertion

- In the AVL tree, the step of insertion is usually followed by an additional step of **rotation**
  - Rotation is done to restore the balance of the tree

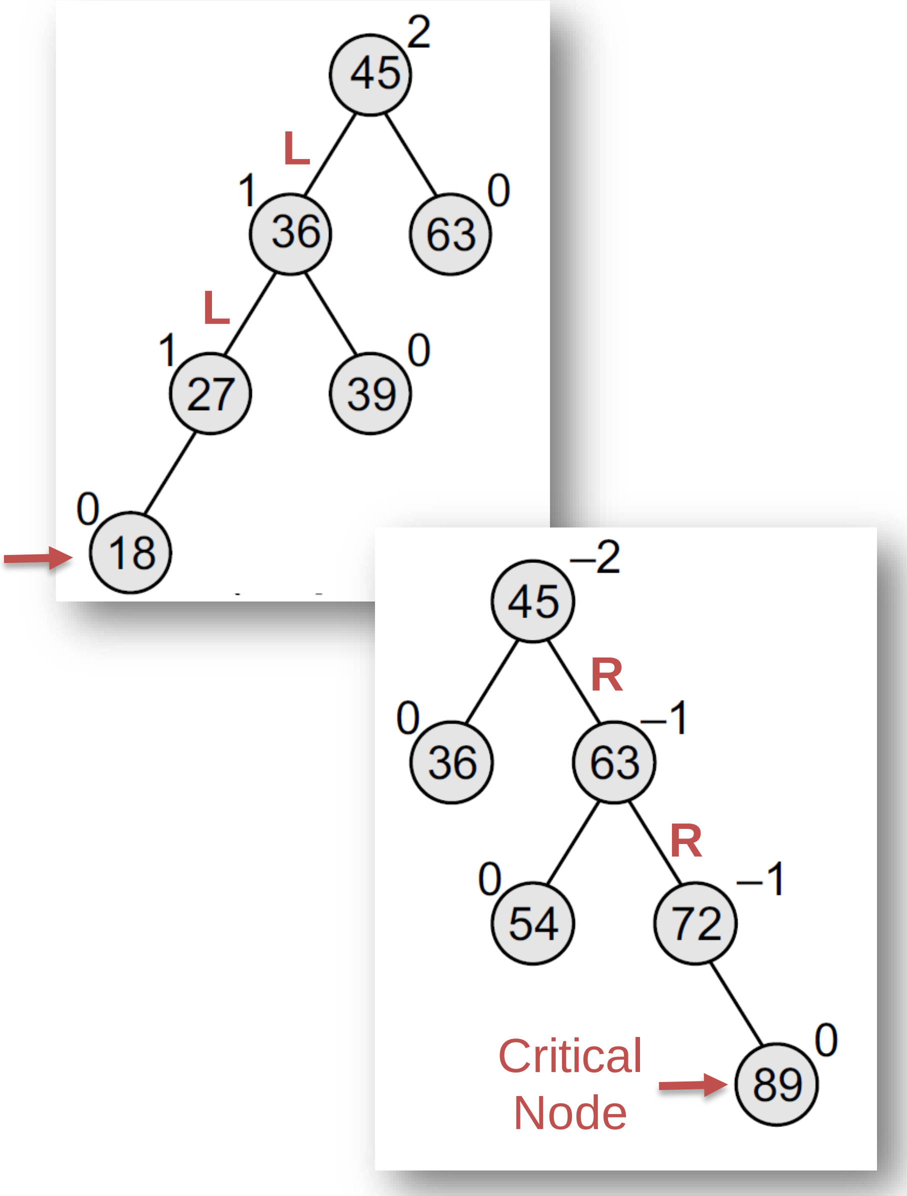
- Insert a node with value 71 in a given AVL tree



Critical Node

Rotations.

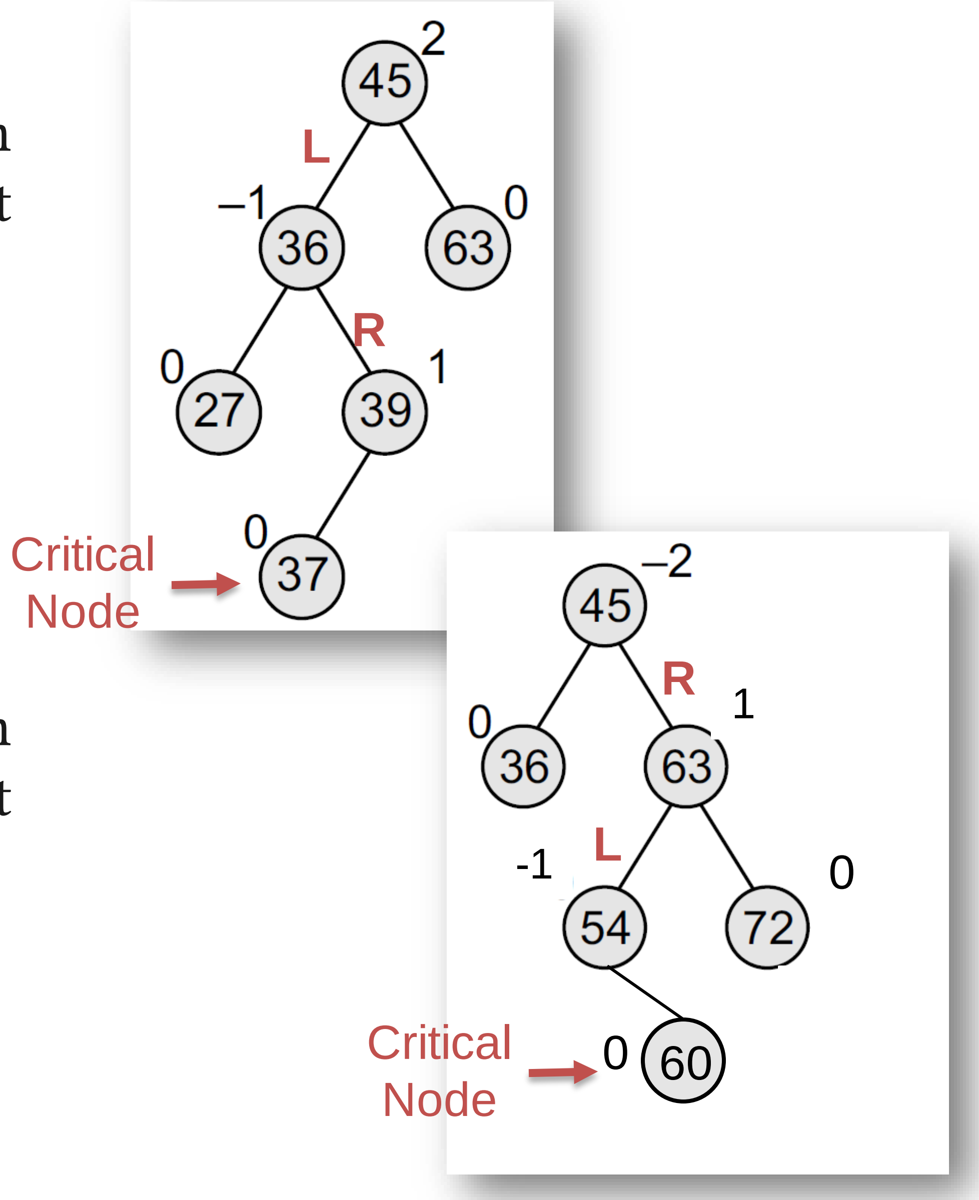
- LL rotation
  - The new node is inserted in the left sub-tree of the left sub-tree
- RR rotation
  - The new node is inserted in the right sub-tree of the right sub-tree



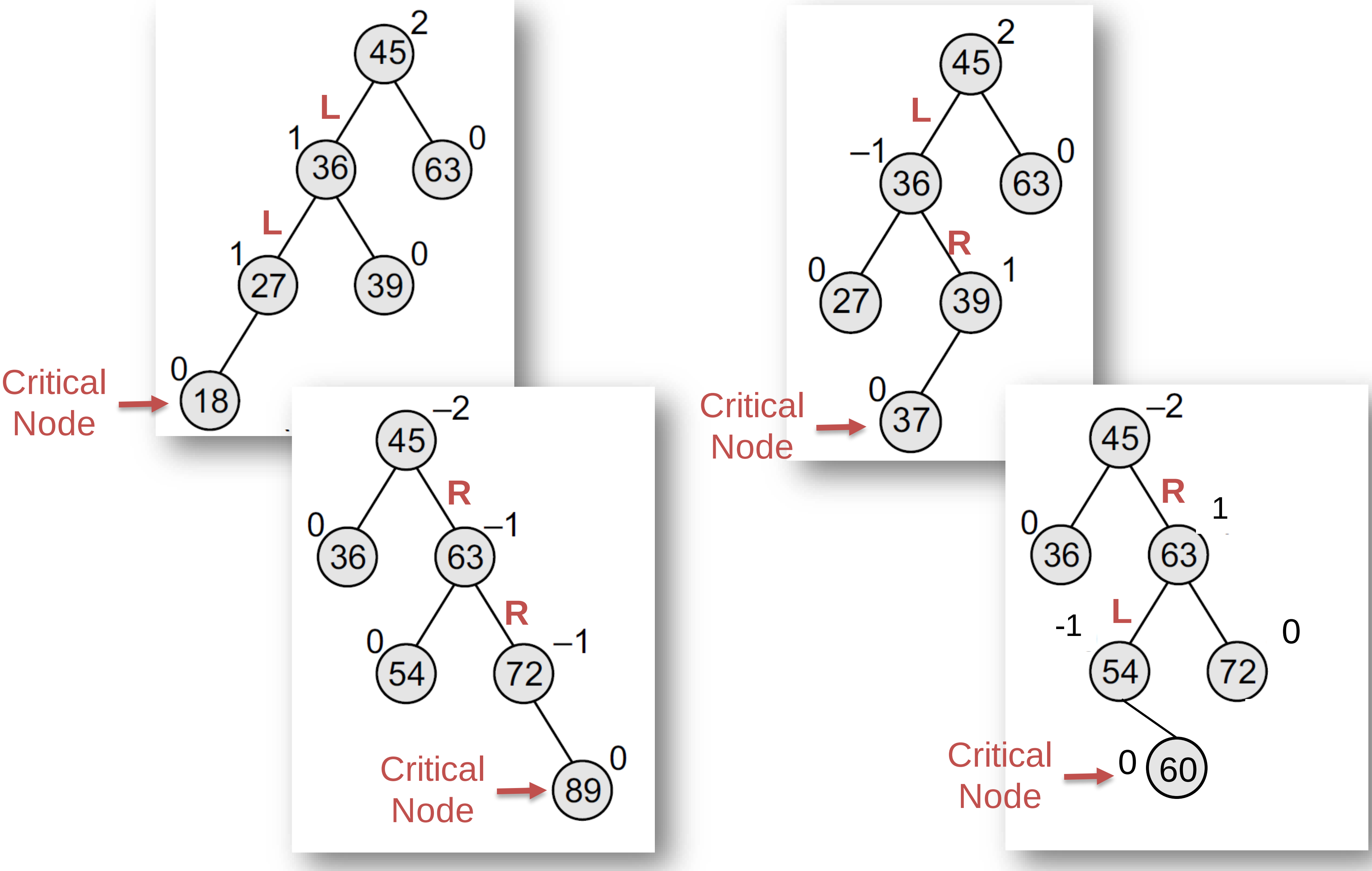
Rotations..

- LR rotation
  - The new node is inserted in the right sub-tree of the left sub-tree

- RL rotation
  - The new node is inserted in the left sub-tree of the right sub-tree



Rotations...



Critical Node

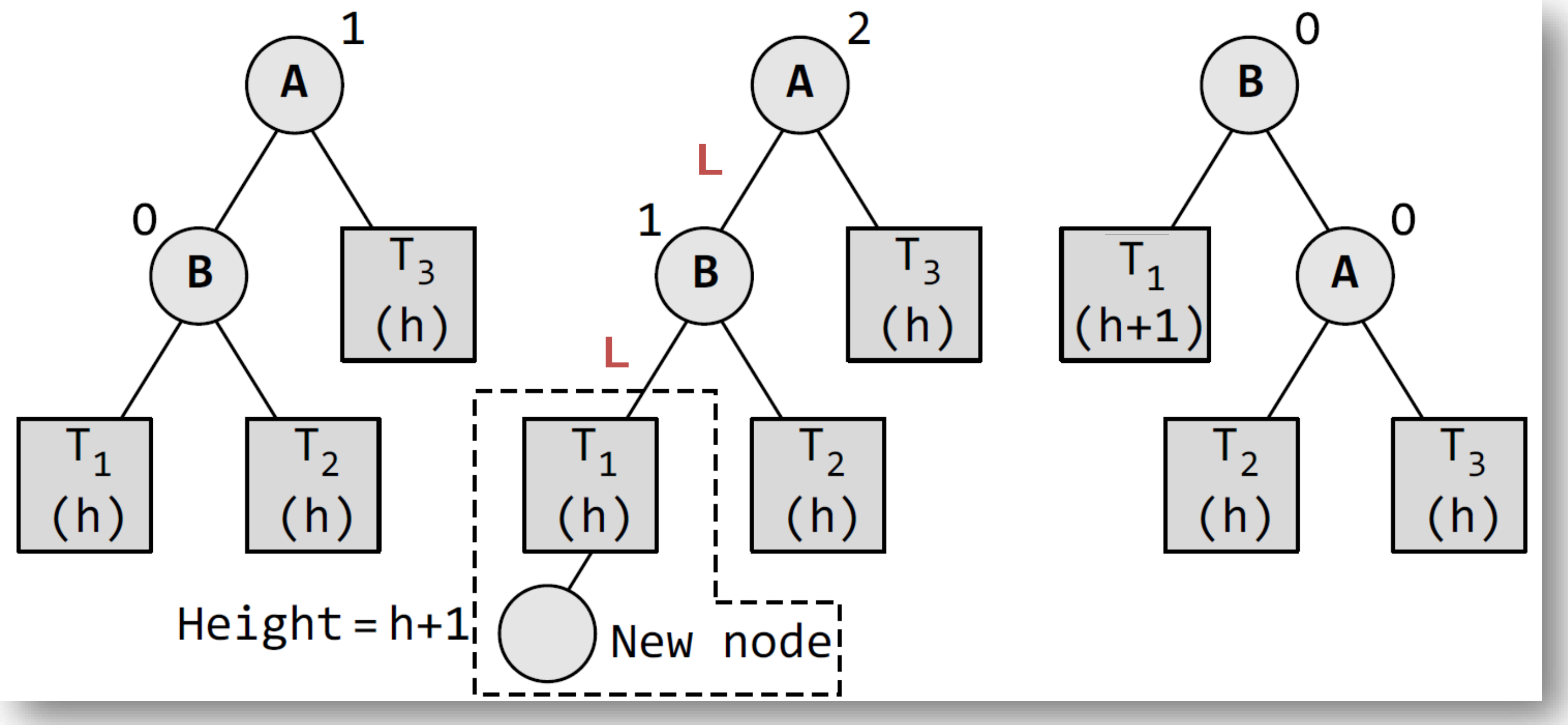
Critical Node

Critical Node

Critical Node

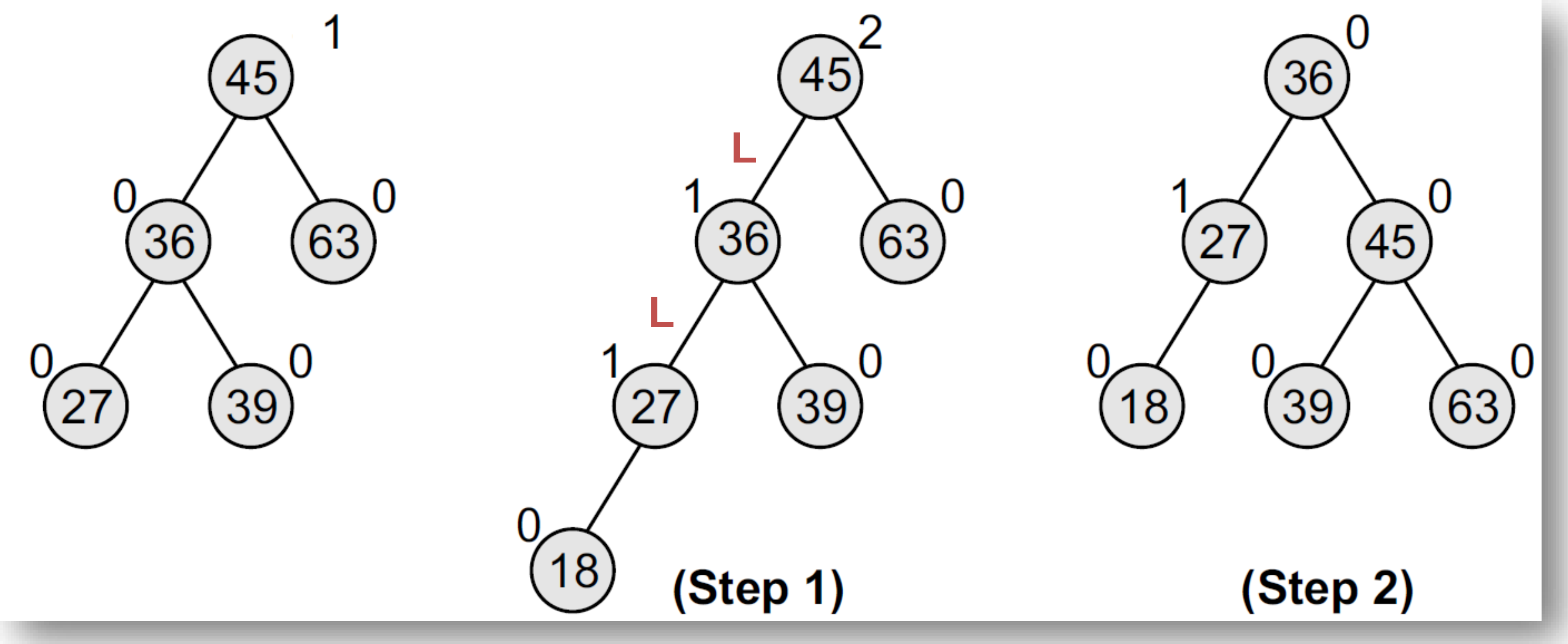
LL Rotation.

- By LL Rotation
  - Node B becomes the root, with T1 and A as its left and right child
  - T2 and T3 become the left and right sub-trees of A



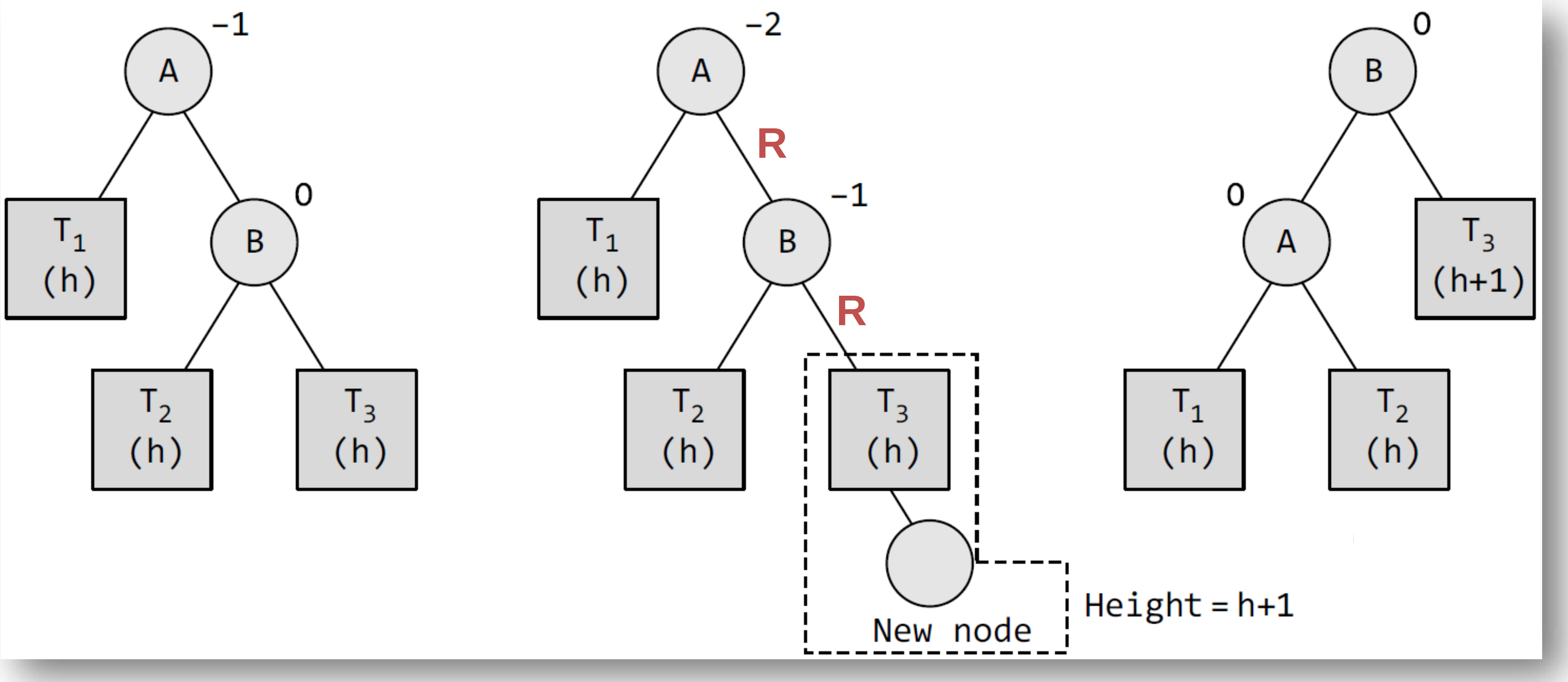
LL Rotation..

- Example for LL Rotation
  - Insert 18 in a given AVL tree



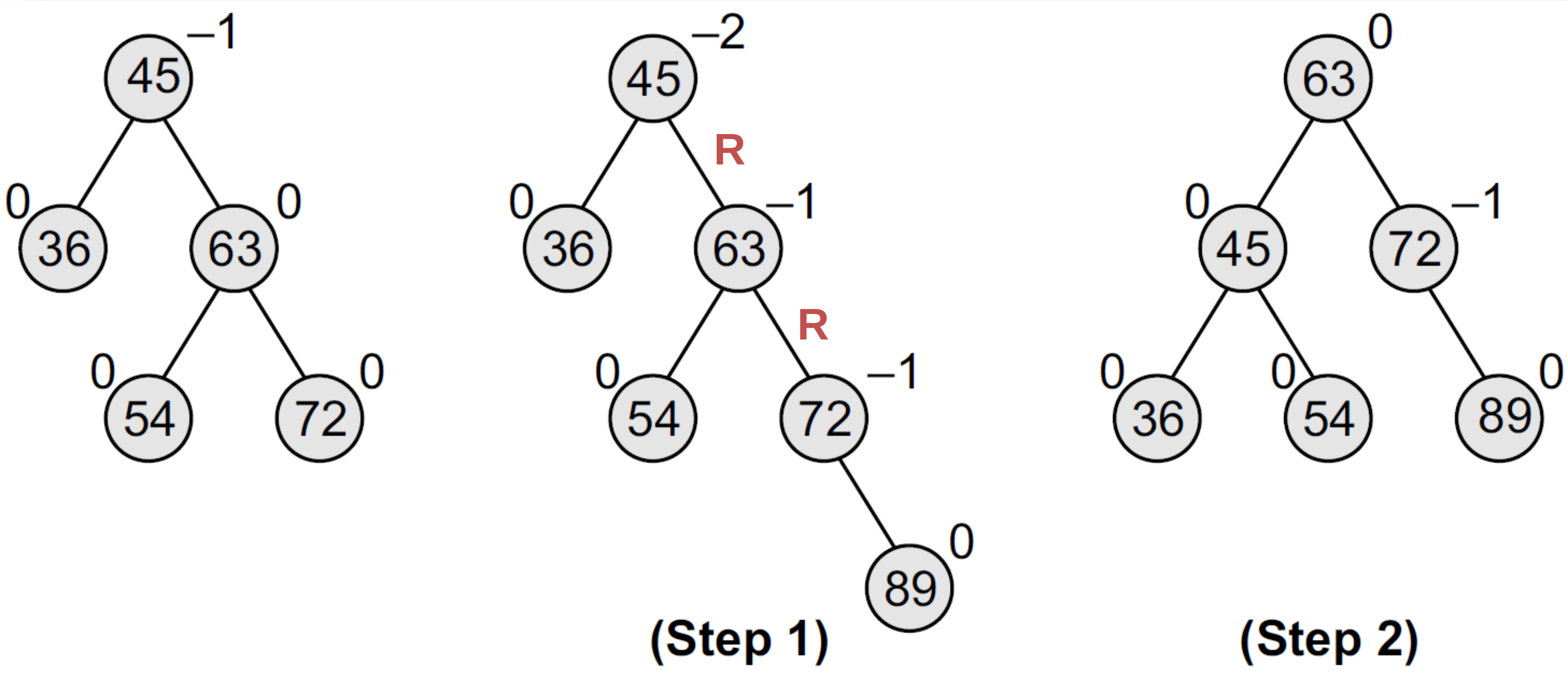
RR Rotation.

- In the context of RR rotation
  - Node B becomes the root, with A and T3 as its left and right child
  - T1 and T2 become the left and right sub-trees of A



RR Rotation..

- Example for RR Rotation
  - Insert 89 in a given AVL tree

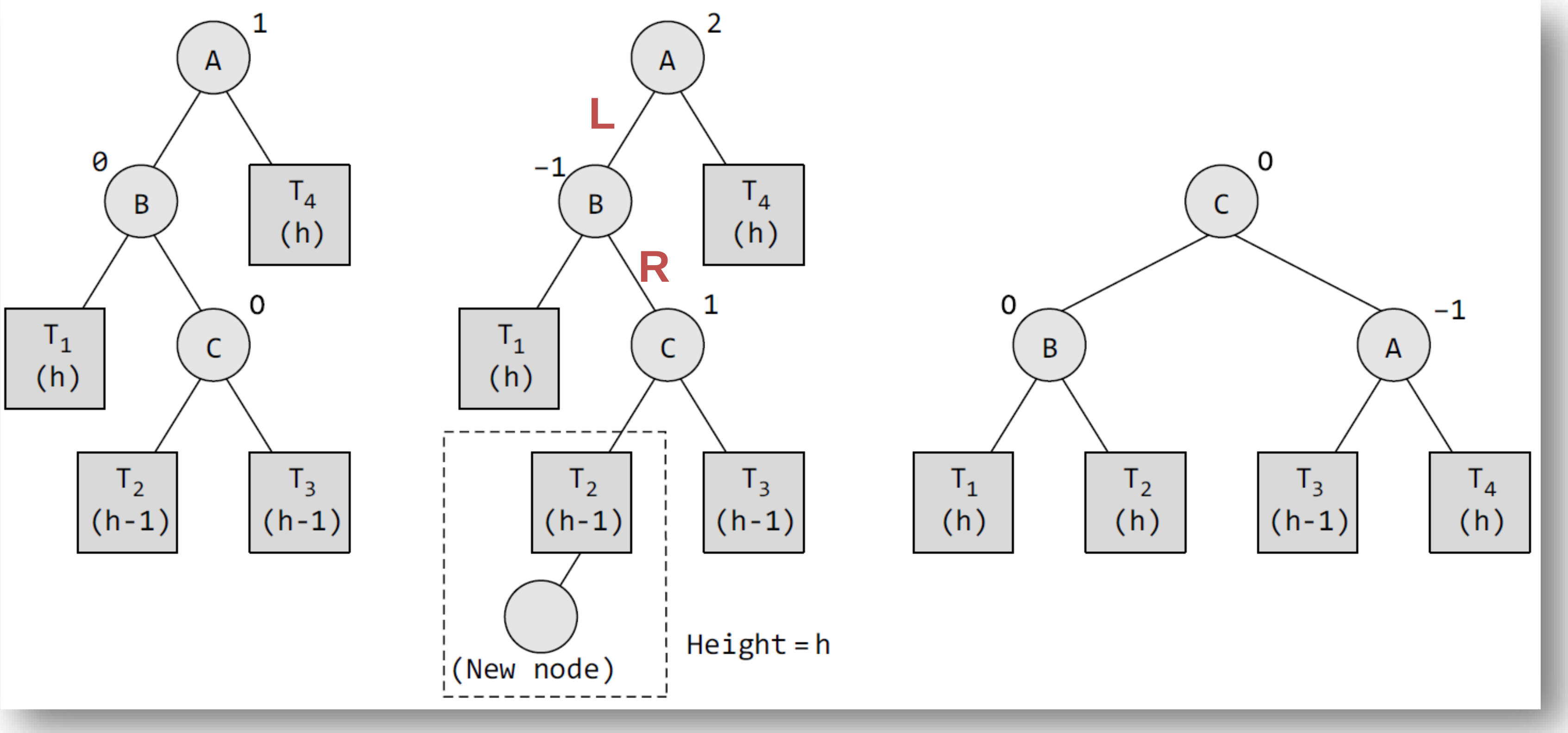


(Step 1)

(Step 2)

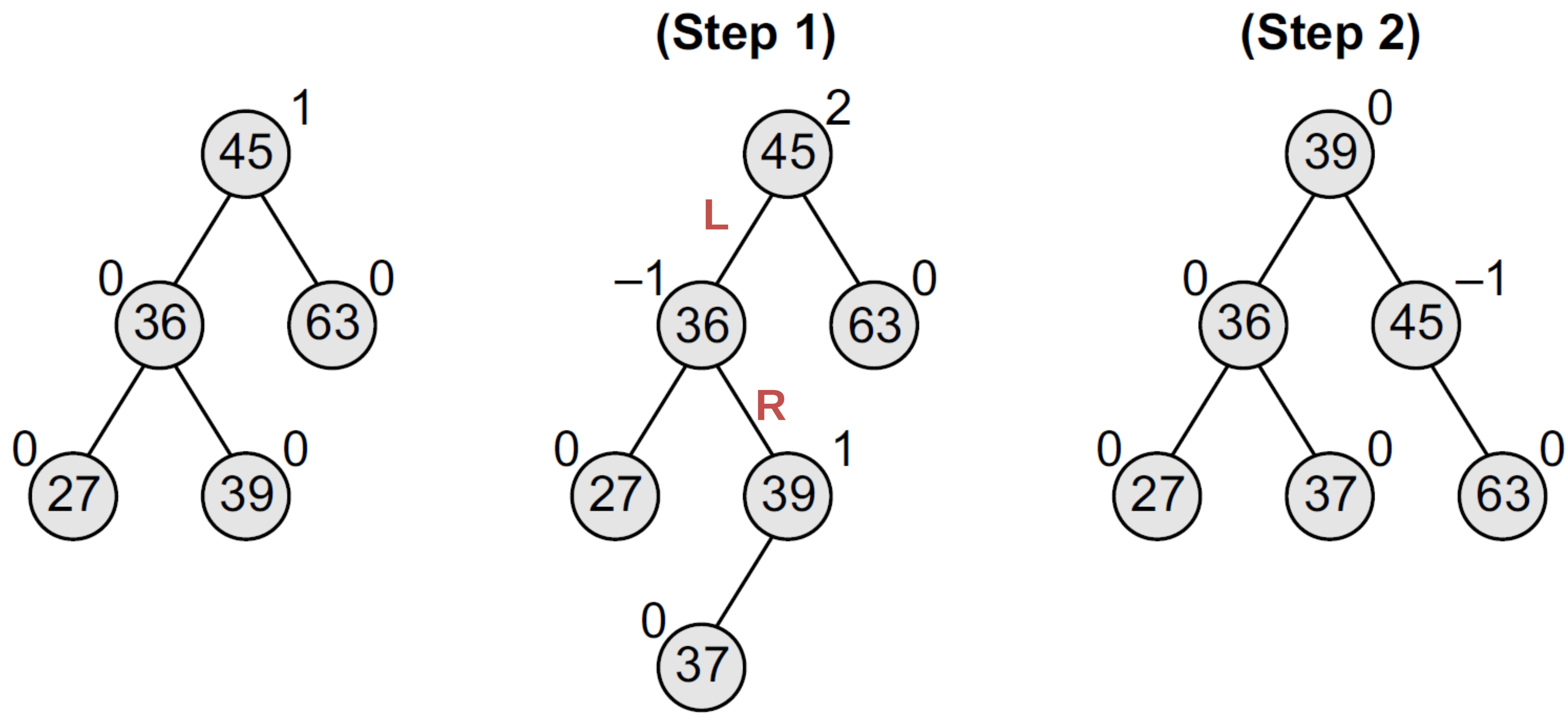
LR Rotation.

- By LR rotation
  - Node C becomes the root, with B and A as its left and right children
  - Node B has T1 and T2 as its left and right sub-trees and T3 and T4 become the left and right sub-trees of node A



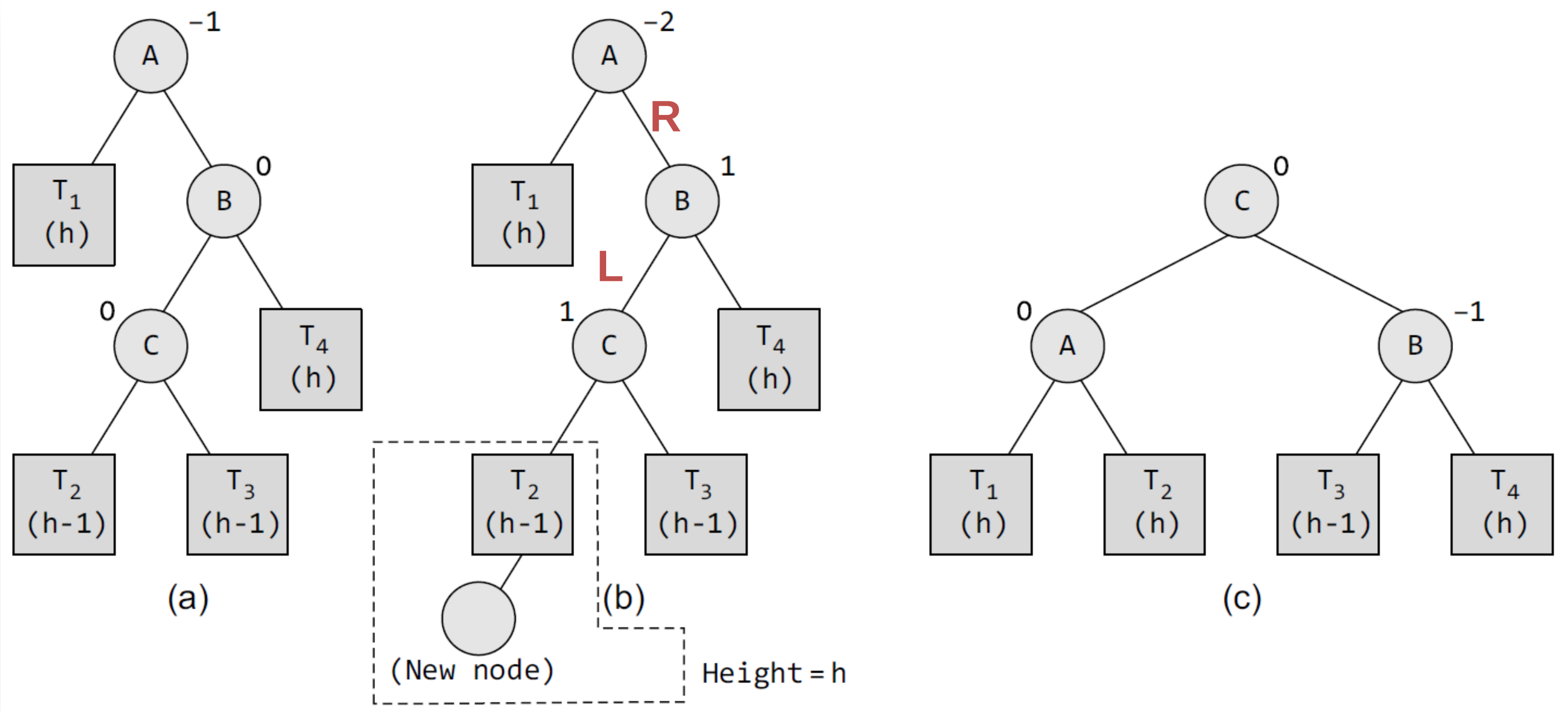
LR Rotation..

- Example for RR Rotation
  - Insert 37 in a given AVL tree



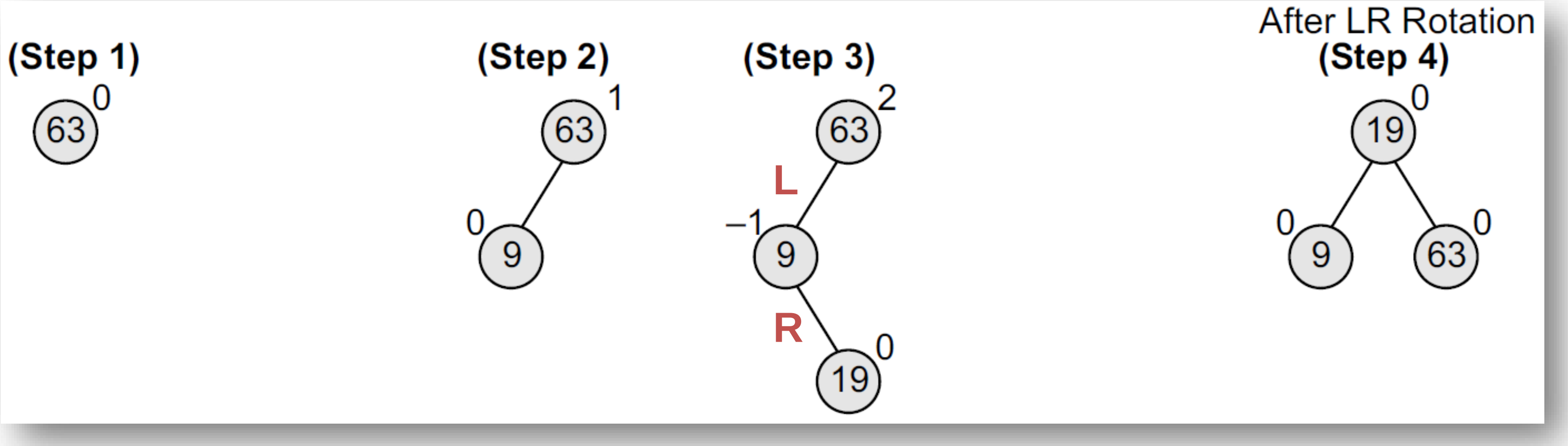
RL Rotation

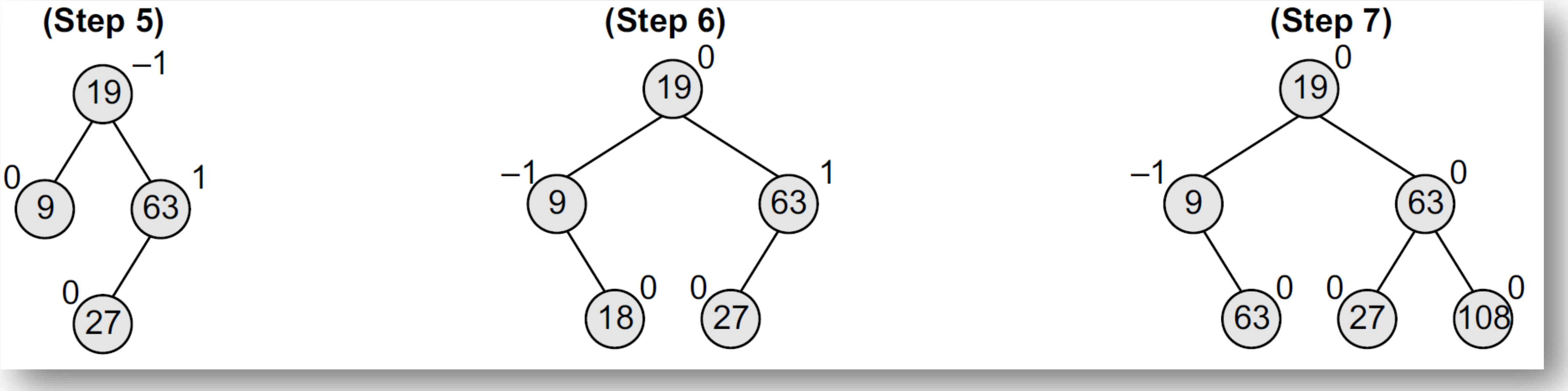
- By RL rotation
  - Node C becomes the root, with A and B as its left and right children
  - Node A has T1 and T2 as its left and right sub-trees and T3 and T4 become the left and right sub-trees of node B



Example.

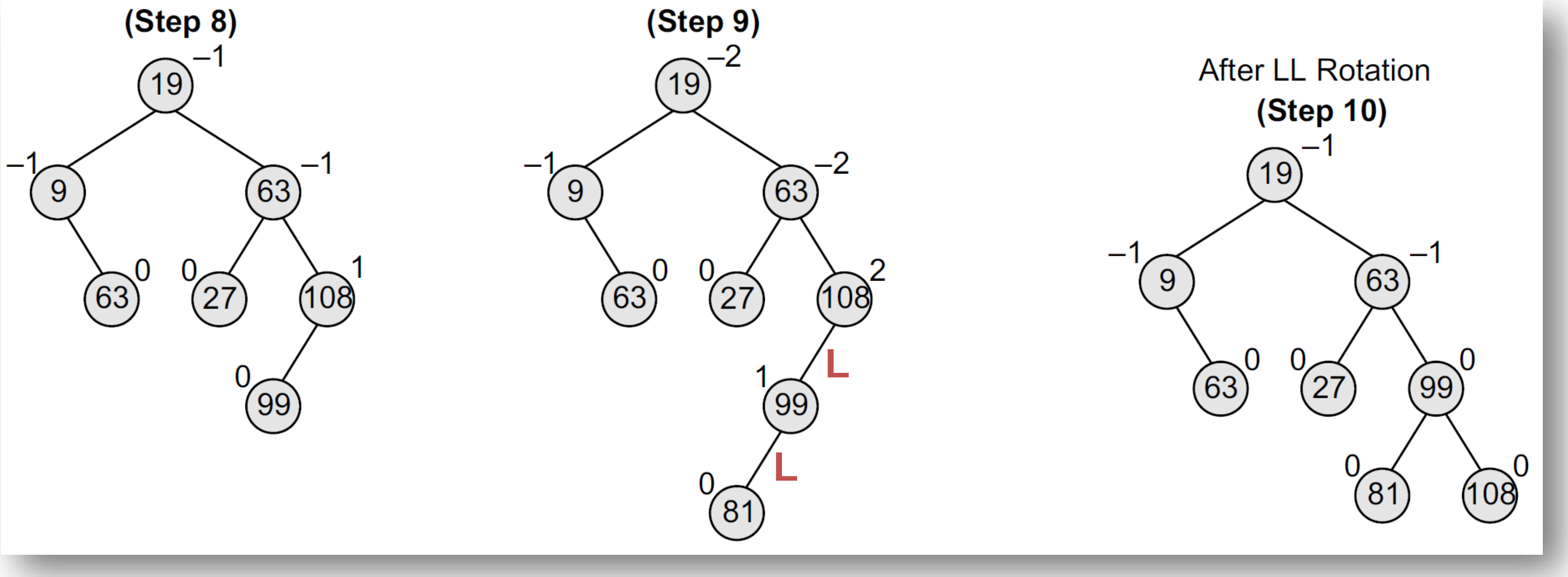
- Construct an AVL tree by inserting the following elements in the given order: 63, 9, 19, 27, 18, 108, 99, 81





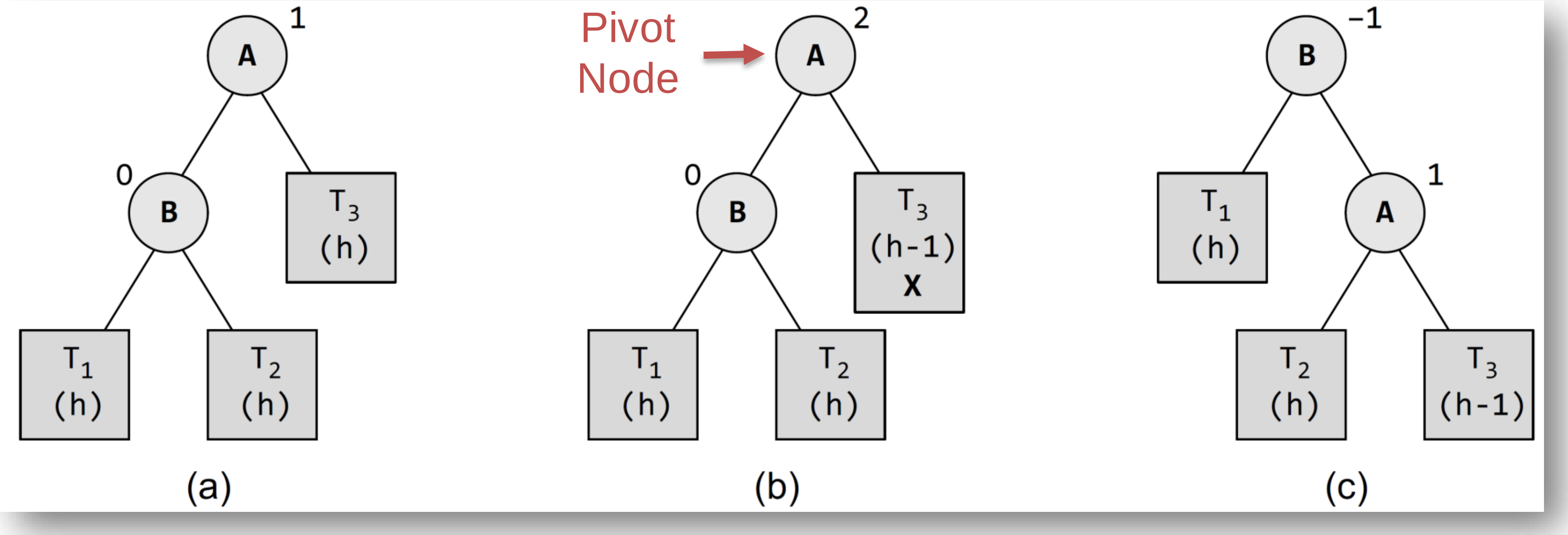
Example..

- Construct an AVL tree by inserting the following elements in the given order: 63, 9, 19, 27, 18, 108, 99, 81



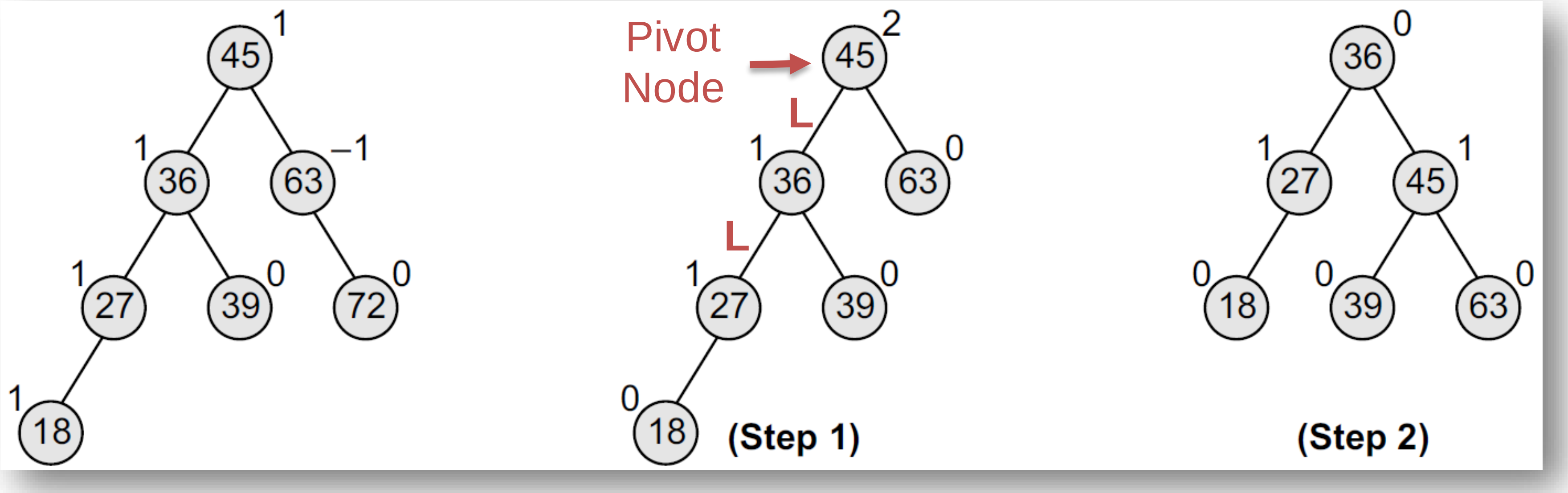
Deletion – R0 Rotation

- Deletion of a node in an AVL tree may disturb the balance of the tree
  - To rebalance the AVL tree, we need to perform rotations!
  - There are two rotation methods
    - R0 Rotation (LL Rotation)
    - R1 Rotation (LR Rotation)



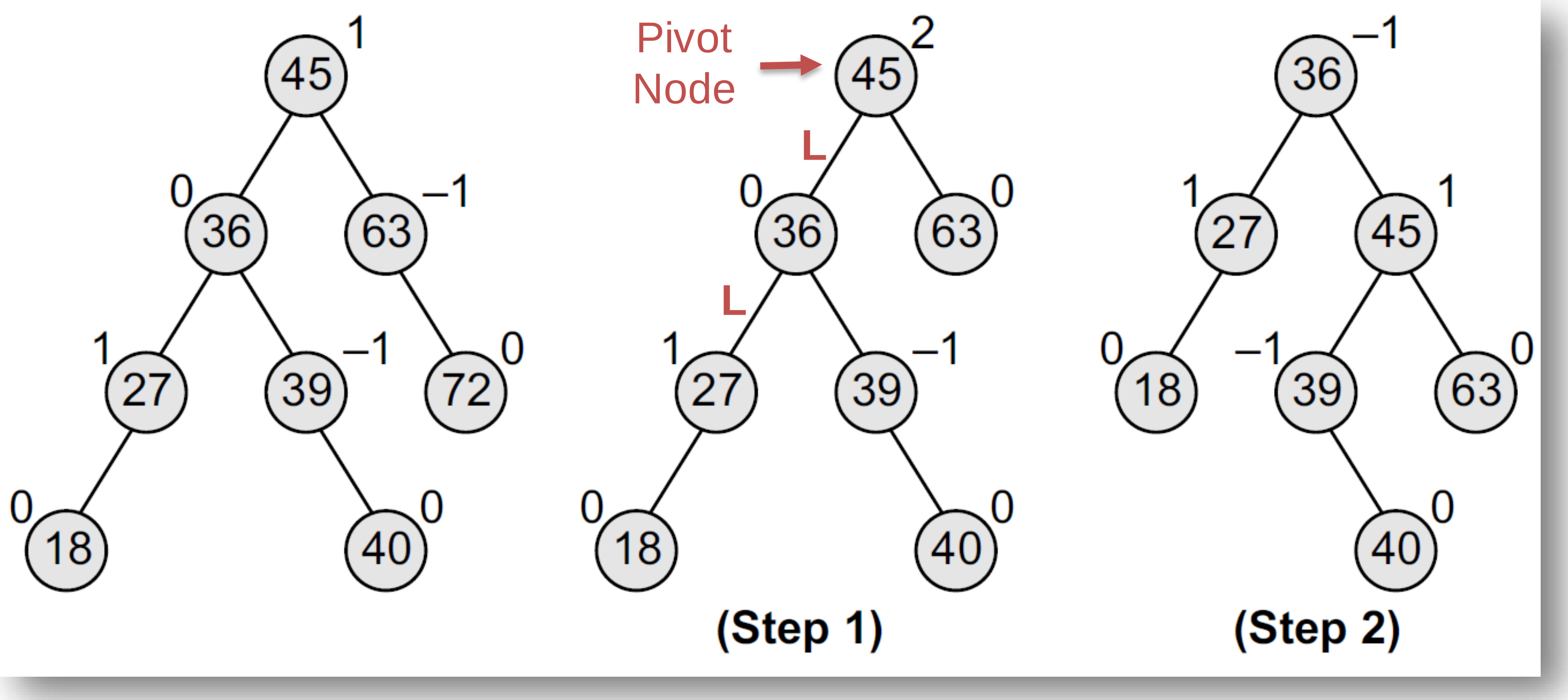
Examples.

- Delete 72 from a given AVL tree



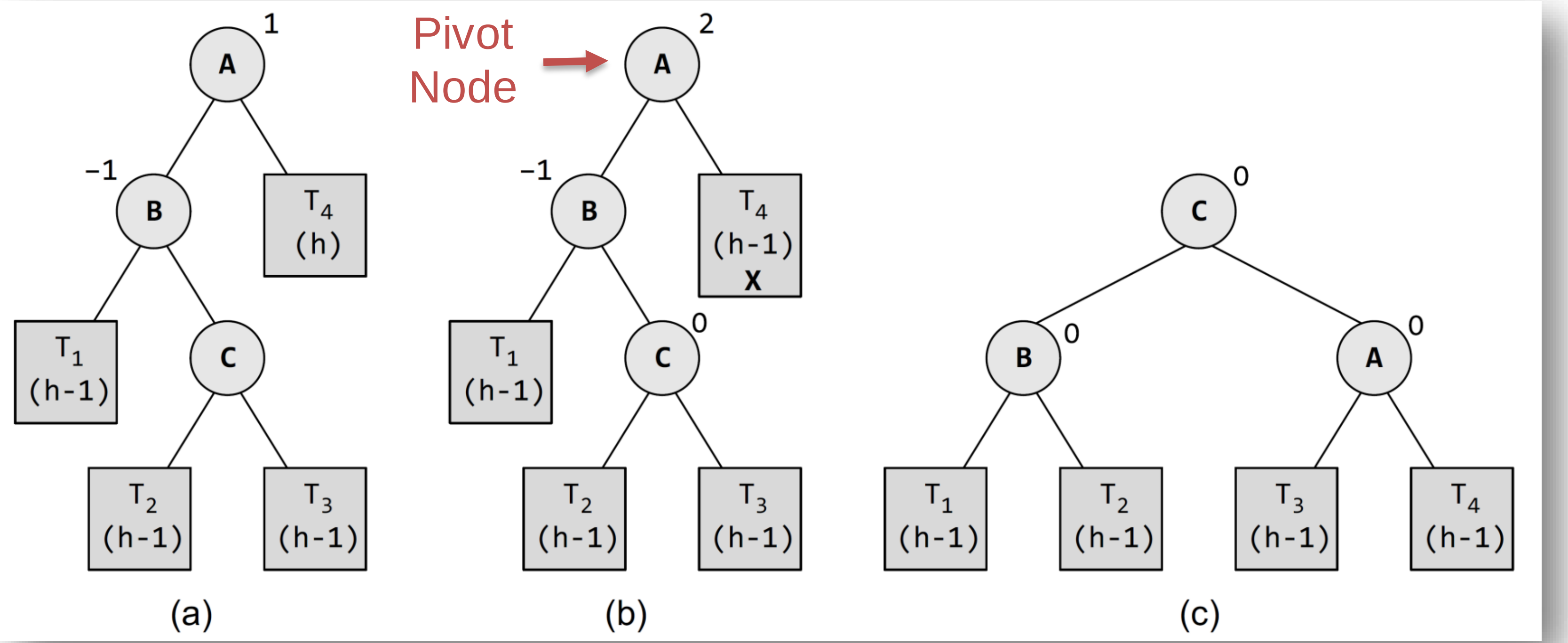
Examples..

- Delete 72 from a given AVL tree



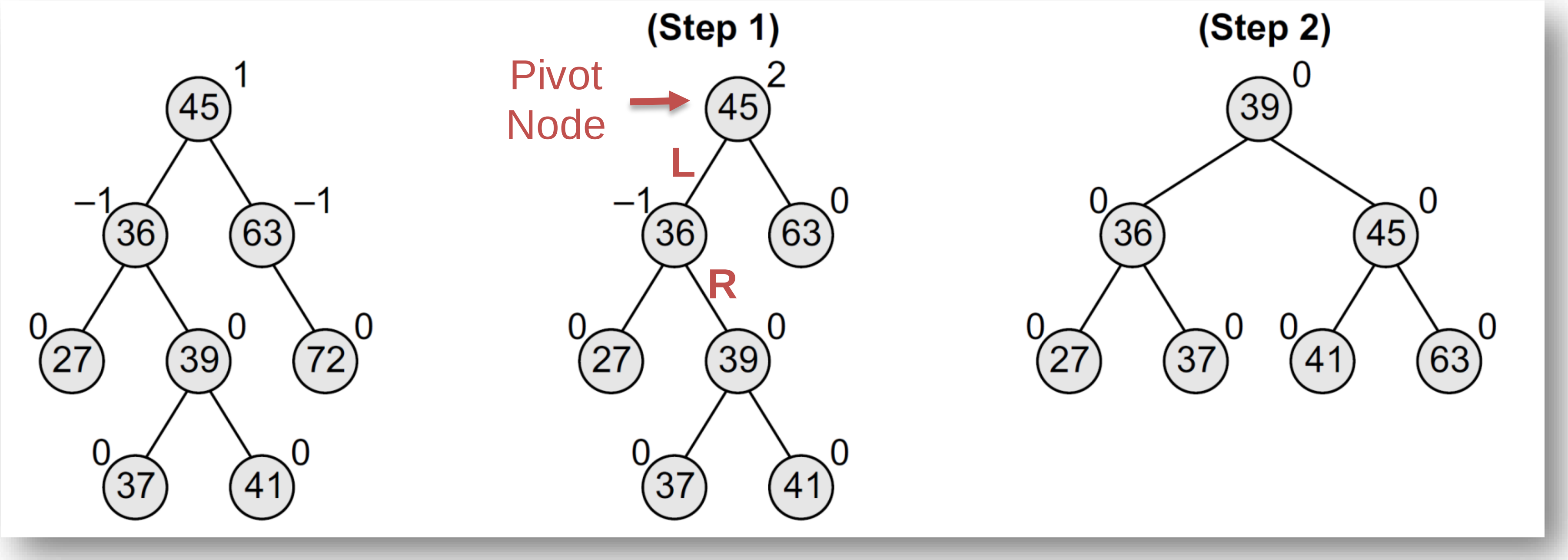
Deletion – R1 Rotation

- Deletion of a node in an AVL tree may disturb the balance of the tree
  - To rebalance the AVL tree, we need to perform rotations!
  - There are four rotation methods
    - R0 Rotation (LL Rotation)
    - R1 Rotation (LR Rotation)



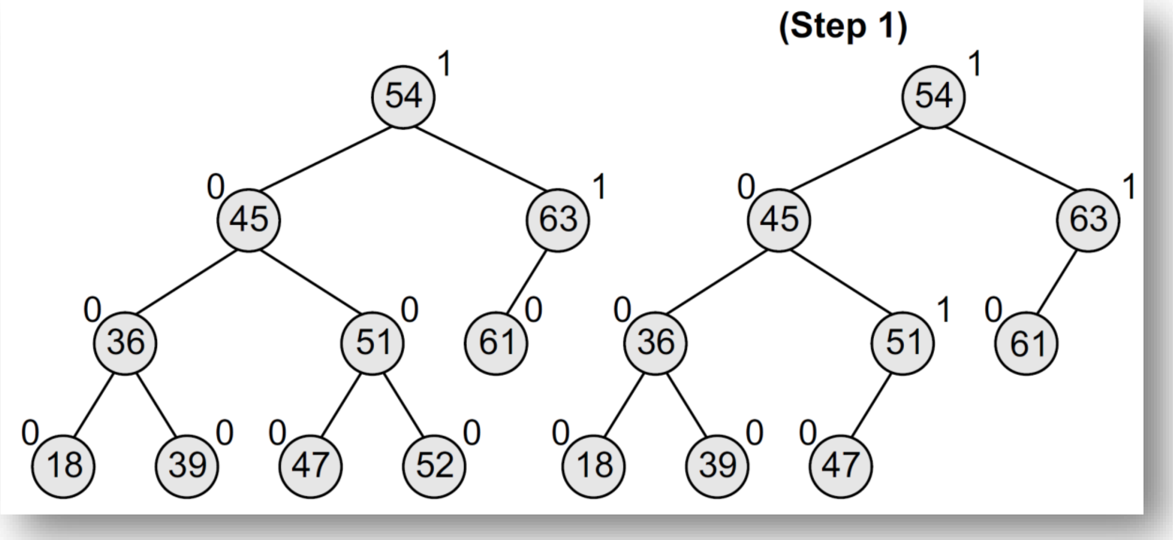
Examples.

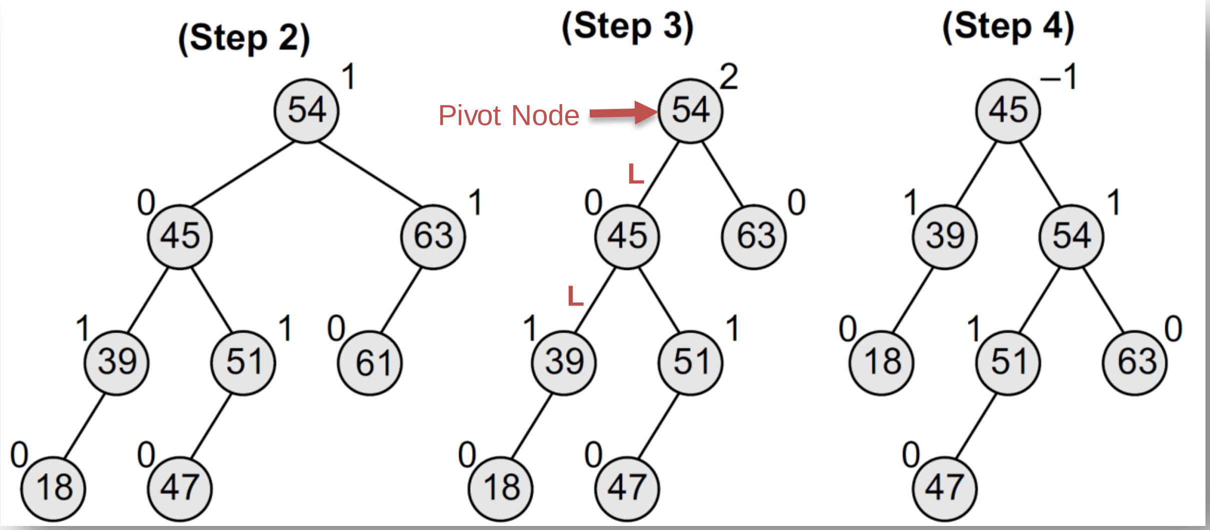
- Delete 72 from a given AVL tree



Examples..

- Delete 52, 36, and 61 from a given AVL tree





Questions?

kychen@mail.ntust.edu.tw