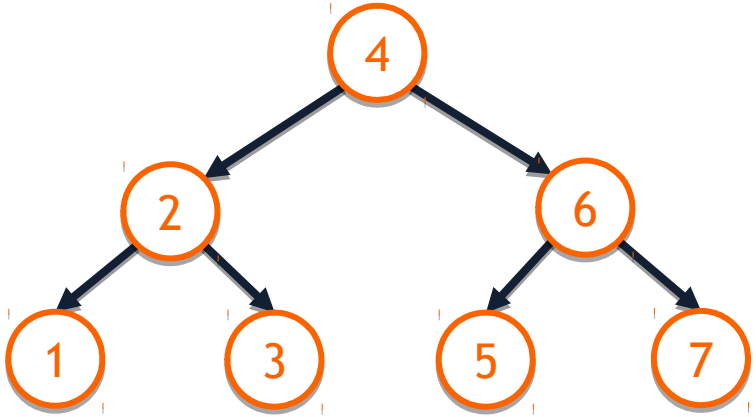
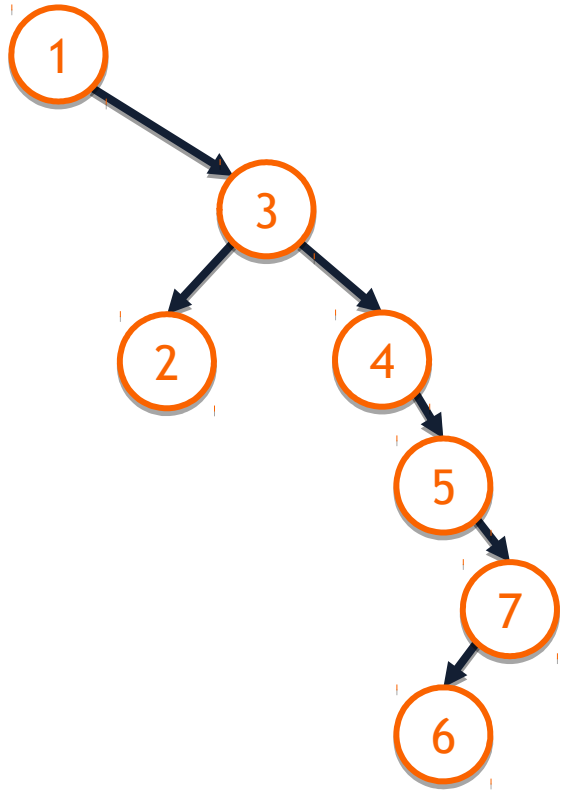
Balanced BST

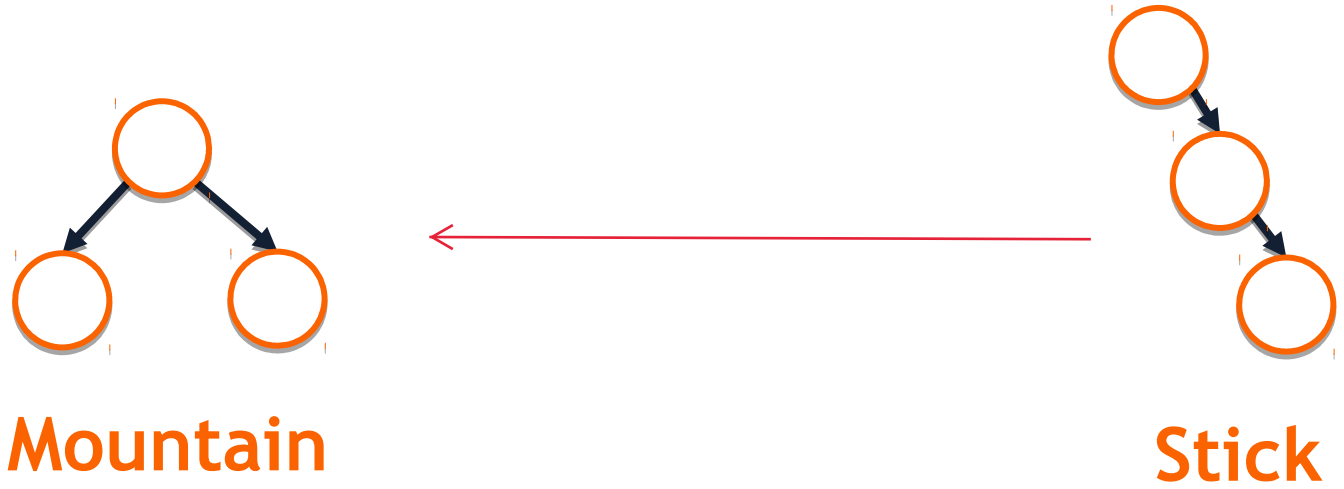
Prof. Wade Fagen-Ulmschneider

Balanced BSTs are height-balanced trees that ensures nearly half of the data is located in each subtree:



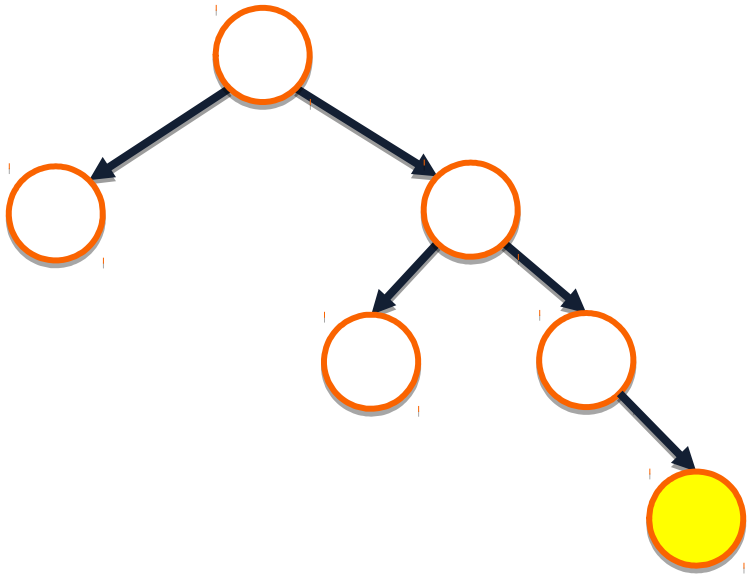


BST Sub-structures



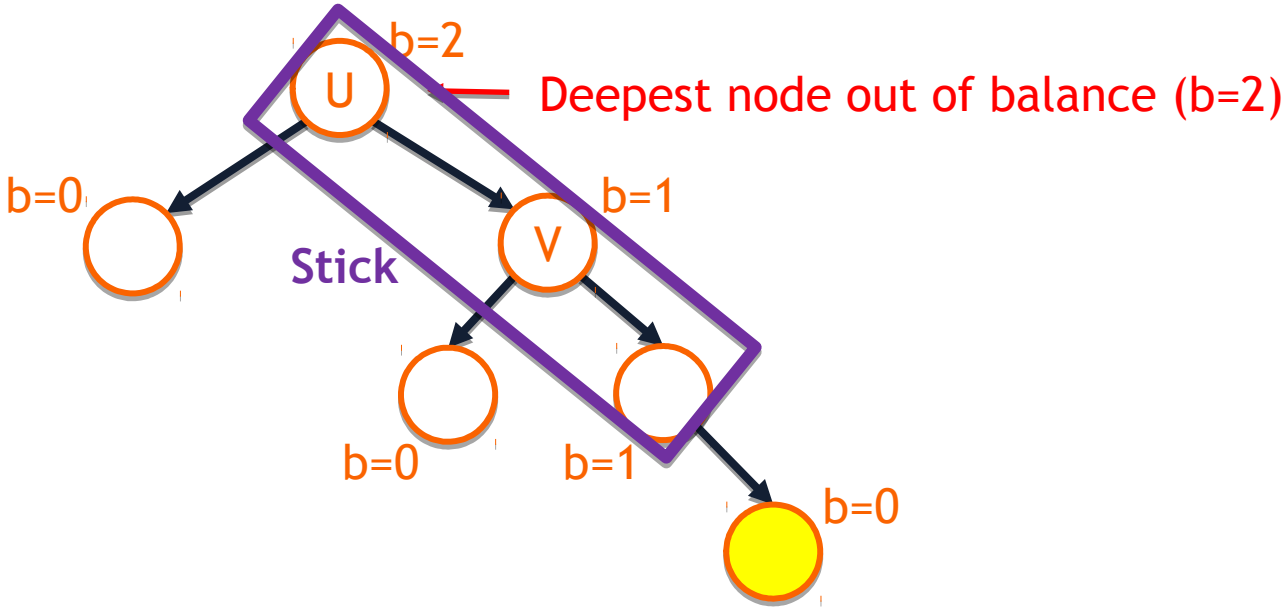
Example: BST Insert

Consider a new node inserted into an initially balanced BST:



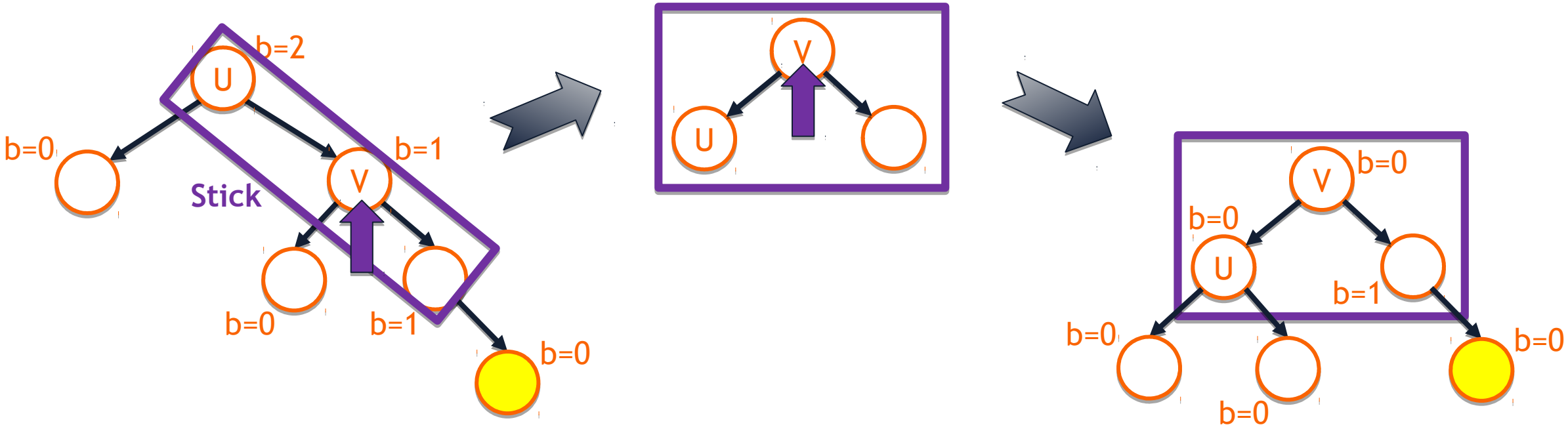
BST Insert

We identify the deepest node in the tree that is out of balance:



BST Rotation

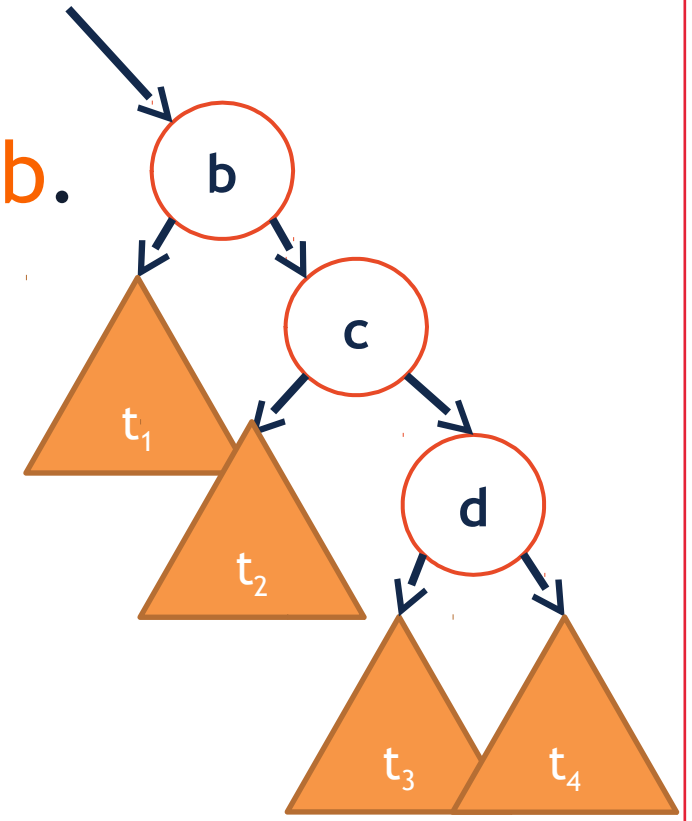
To turn a stick into a mountain: Break the stick in half, raise it up as a mountain, and re-attach the children:



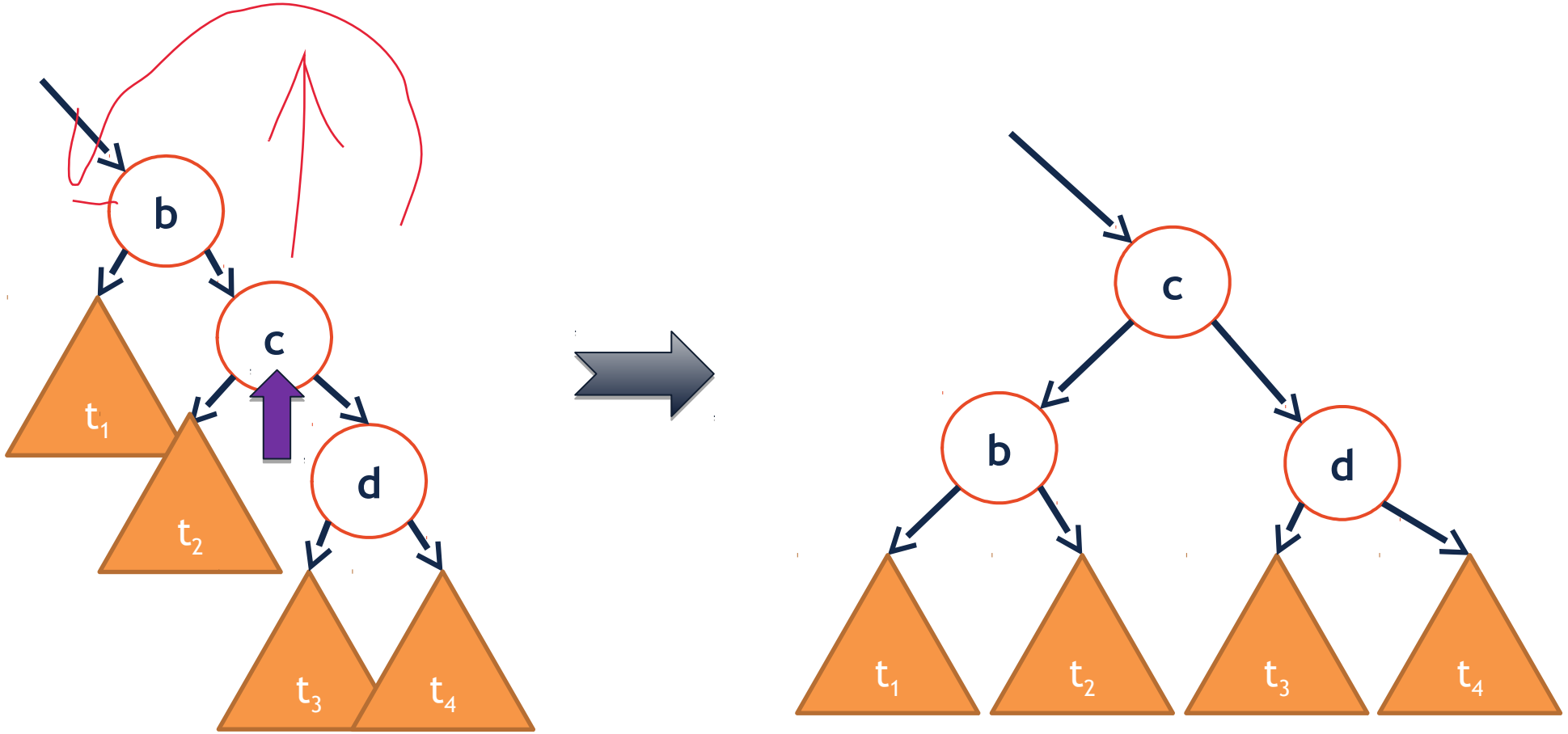
Generic Left Rotation

Consider an arbitrary tree such that:

- The deepest point of imbalance is at node b.
- The balance factor of b is **2**.
- The balance factor of c is **1**.
- An insert in t_3 or t_4 caused the imbalance.

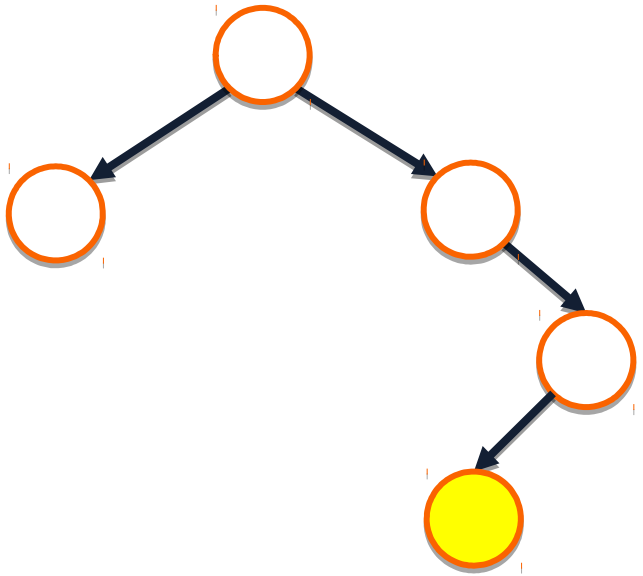


Generic Left Rotation



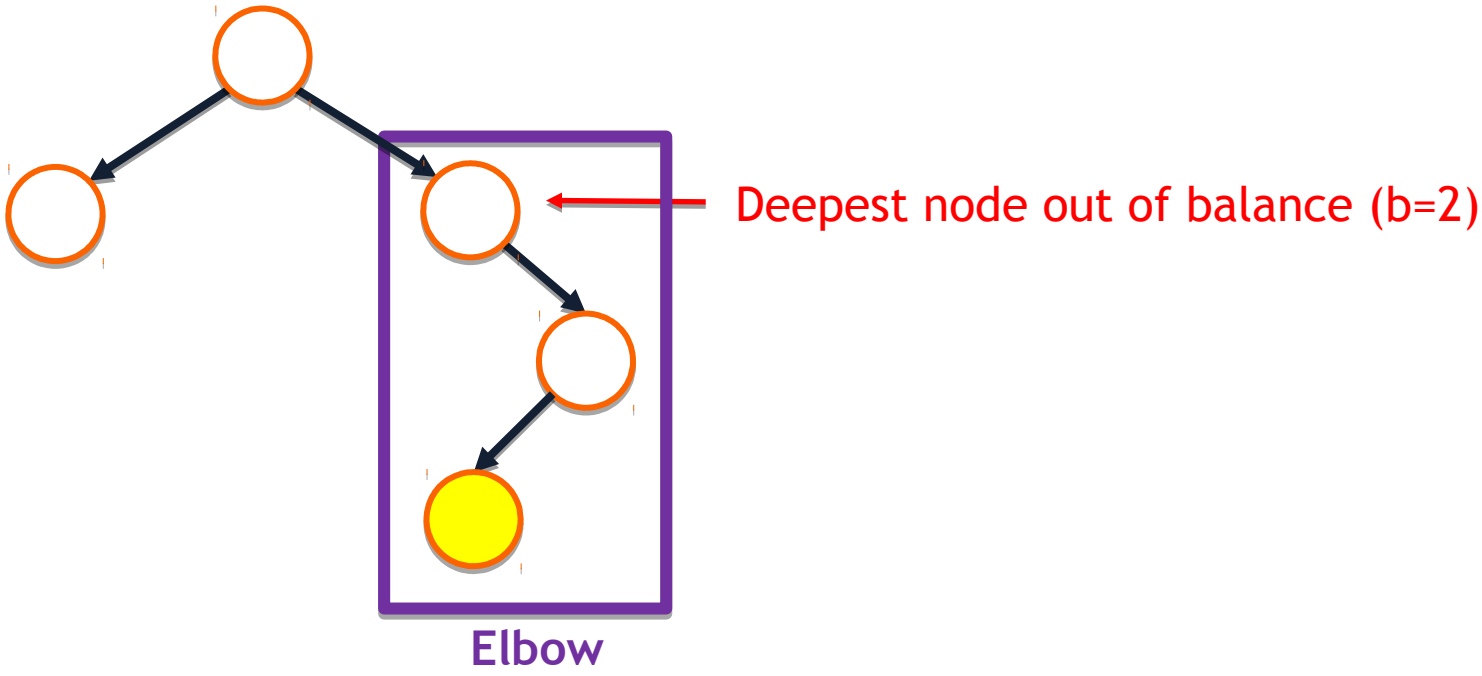
BST Insert, Example #2

Consider a new node inserted into an initially balanced BST:



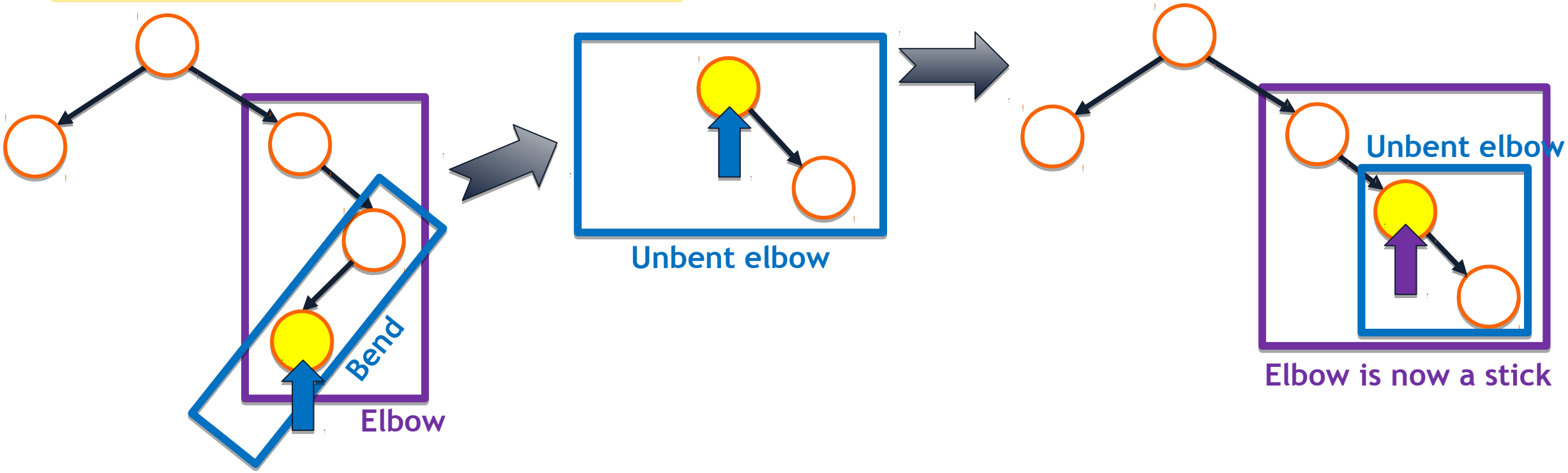
BST Insert

We identify the deepest node in the tree that is out of balance:



BST Rotation

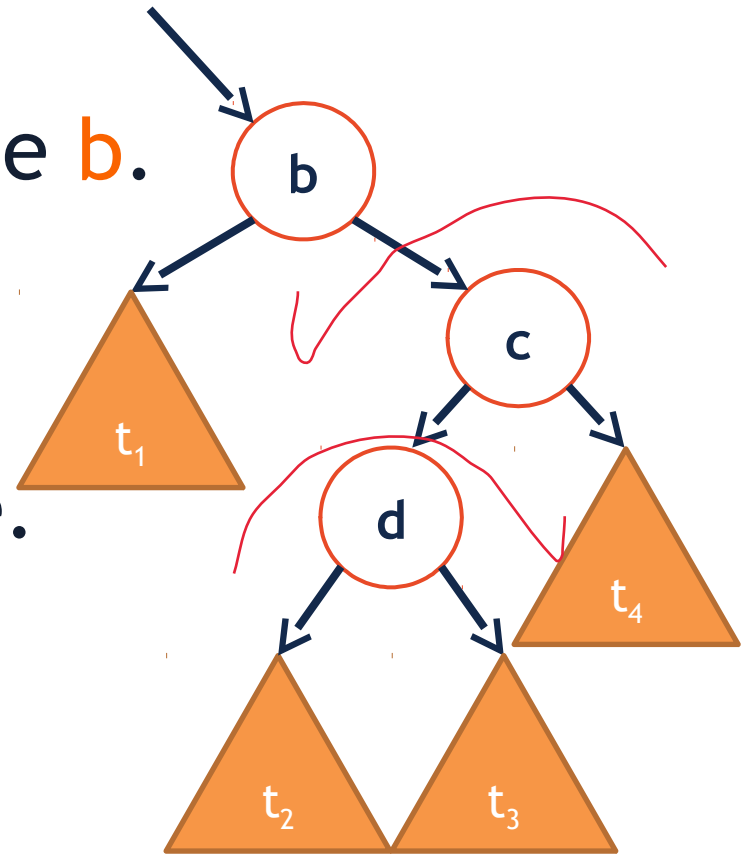
We “unbend” the elbow with a rotation about the bend and then we have a stick:



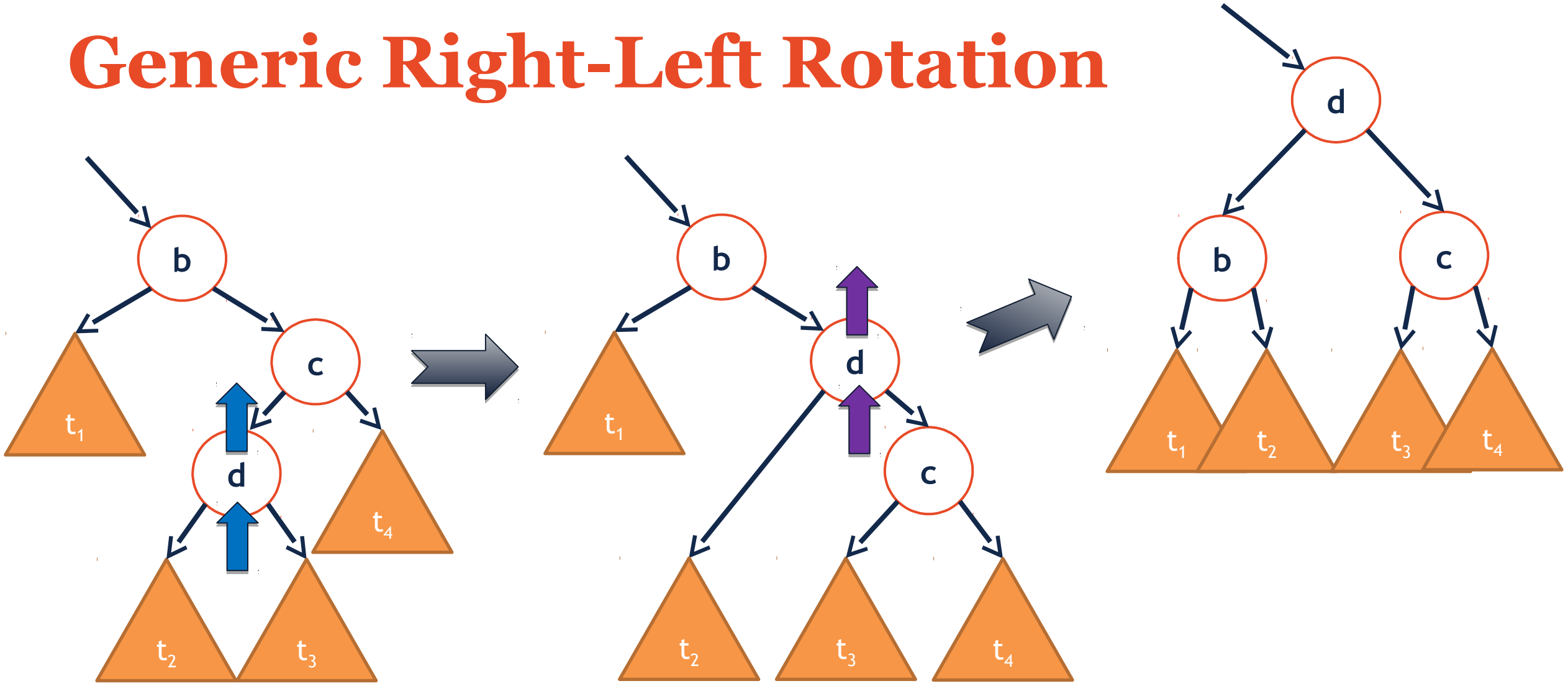
Generic Right-Left Rotation

Consider an arbitrary tree such that:

- The deepest point of imbalance is at node b.
- The balance factor of b is **2**.
- The balance factor of c is **-1**.
- An insert in t_2 or t_3 caused the imbalance.



Generic Right-Left Rotation



Rotation Summary

	Balance factor (b) of the lowest point of imbalance	Balance factor (b) of the node in the direction of imbalance
Left Rotation	2	1
Right Rotation	-2	-1
Right-Left Rotation	2	-1
Left-Right Rotation	-2	1

BST Rotations

- BST rotations restore the balance property to a tree after an insert causes a BST to be out of balance.

- Four possible rotations: L, R, LR, RL
  - Rotations are local operations.
  - Rotations do not impact the broader tree.
  - Rotations run in O(1) time.

- These trees are called "AVL Trees"