

EECS3101 (Section E) Fall 2025  
Tutorial: Week 4  
Rotations in Self-Balancing BSTs

Chen-Wei Wang

Release Date: Friday, September 26

**No Submission Required: Complete for Learning & Test Prep**

## Contents

1	On Paper: Performing Trinode Restructuring Steps after Deletions	2
2	In Eclipse: Performing A Single, Right Rotation	2

# 1 On Paper: Performing Trinode Restructuring Steps after Deletions

**Task 0a.** Review the algorithm of BST deletions [here](#).

**Task 0b.** Review the algorithm of performing BST rotations after deletions [here](#).

**Task 1.**

- **Insert** the following sequence of nodes into an empty BST:

$\langle 44, 17, 62, 32, 50, 78, 48, 54, 88 \rangle$

Is the *height-balance property* satisfied?

- **Delete** 32 from the BST.

Is the *height-balance property* maintained?

- Perform the necessary **trinode restructuring step(s)** on the appropriate node(s).

Is the *height-balance property* re-established?

**Task 2.**

- **Insert** the following sequence of nodes into an empty BST:

$\langle 50, 25, 10, 30, 5, 15, 27, 1, 75, 60, 80, 55 \rangle$

Is the *height-balance property* satisfied?

- **Delete** 80 from the BST.

Is the *height-balance property* maintained?

- Perform the necessary **trinode restructuring step(s)** on the appropriate node(s).

Is the *height-balance property* re-established?

# 2 In Eclipse: Performing A Single, Right Rotation

- Download EECS3101\_Tutorials.Week.04\_Starter.zip from [here](#).
- Study the two methods `setLeft` and `setRight` in `BSTNode.java` (under the `model` package).
- Study the requirements as specified in `TestBST.java` (under the `tests` package).
- Implement the two methods in `BSTUtilities.java` (under the `model` package):

```
– public ArrayList<BSTNode<E>> inOrderTraversal(BSTNode<E> root)
```

```
– public void rightRotate(BSTNode<E> a, BSTNode<E> b, BSTNode<E> c)
```