

Lab 3

Release: 20 Oct 2023 (Fri, Week 10)

Due: Online submission in eLearn by Sunday, 29 Oct 2023, 11pm

Instructions:

1. This is an individual lab. You will be awarded class participation points for the labs component based on the completion of submission, not on the efficiency of solution (you are encouraged to explore the most efficient approach for your own learning purposes)
2. You can discuss the general approach to solve the question but you should not be sharing your codes with anyone.
3. Unless specifically stated in the question, you are allowed to create additional methods in the classes.
4. Please ensure that all your codes can be compiled.
5. Multiple submissions are allowed but only the most recent submission will be kept in the system as the final submission.
6. Please zip up your submission as named the zip file as <StudentName>.zip.
7. The deadline is final, late submissions will not be accepted.

Q1 – Verify traversals of a binary search tree

Given 3 traversals, the objective is to verify if the 3 traversals are a valid combination of a binary search tree. Compile and run Q1Test.java to test your implementation.

- Write a method named **verify** that takes in 3 strings parameter (String traversal1, String traversal2, String traversal3) containing the traversal of a binary search tree.
- The input parameters could be of any sequence e.g. traversal1 could be Inorder, Preorder, Postorder
- The nodes in the sequence are separated by “-” and contains integers only e.g. (1-2-3)
- The nodes are unique in a traversal. e.g. (there won’t be 1-2-2-3)
- If the 3 traversals are valid, the output prints the traversal orders of the respective strings. Otherwise, prints “Invalid traversals”
- You can include additional classes.

Traversal 1 : 1-2

Traversal 2 : 2-1-3

Traversal 3 : 1-3-2

Expected Result : Invalid traversals

Actual Result : Invalid traversals

Traversal 1 : 1-2-3

Traversal 2 : 2-1-3

Traversal 3 : 1-3-2

Expected Result : Traversal 1 - Inorder, Traversal 2 - Preorder, Traversal 3 - Postorder

Actual Result : Traversal 1 - Inorder, Traversal 2 - Preorder, Traversal 3 - Postorder

Traversal 1 : 1-2-3

Traversal 2 : 3-2-1

Traversal 3 : 2-3-1

Expected Result : Invalid traversals

Actual Result : Invalid traversals

Traversal 1 : 3-1-2-5-4

Traversal 2 : 2-1-4-5-3

Traversal 3 : 1-2-3-4-5

Expected Result : Traversal 1 - Preorder, Traversal 2 - Postorder, Traversal 3 - Inorder

Actual Result : Traversal 1 - Preorder, Traversal 2 - Postorder, Traversal 3 - Inorder

Traversal 1 : 10-20-30-40-50

Traversal 2 : 20-10-30-40-50

Traversal 3 : 10-50-40-30-20

Expected Result : Traversal 1 - Inorder, Traversal 2 - Preorder, Traversal 3 - Postorder

Actual Result : Traversal 1 - Inorder, Traversal 2 - Preorder, Traversal 3 - Postorder