

Assignment 4

1.

Within a binary search tree, a predetermined target value has been specified. You are asked to: a) print the path(s) where target value is present

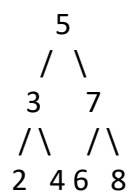
b) find the next greater element of the target value in the binary search tree.

c) implement **recursive** methods for inorder, preorder and postorder traversal

d) compare the arrays generated from the preorder, inorder, and postorder traversals.

Identify the elements that match at the same positions across any two arrays. Print the matching elements along with their indices.

Input:



a) Target: 3

Path(s) : 5 – 3 – 2

5 – 3 – 4

b) Output: 4 is next greater element of the target value

c)

Preorder: 5, 3, 2, 4, 7, 6, 8

Postorder: 2, 4, 3, 6, 8, 7, 5

Inorder: 2, 3, 4, 5, 6, 7, 8

d)

Preorder - Inorder: Match at indices 1 and 6 - elements 3 and 8.

Preorder - Postorder: No elements match at the same index.

Inorder - Postorder: Match at indices 0 and 5 - elements 2 and 7.

NOTE You can use the above examples to test your code. External collection libraries are not allowed

2.

Write a Java program that reads a text file named **hashtable.txt**, processes its content word by word, and counts the occurrences of each word. Finally, the program should display the word frequencies in the format word: count.

Input- Attached text file named hashtable.txt.

Output-A list of words and their frequencies in the format word: count.

Words should be treated as case-insensitive ("Hello" and "hello" are the same word)

NOTE External collection libraries are not allowed. Implement Hash table DS from the scratch. You can use I/O libraries.