Data Structure and Programming, Fall 2018
Programming Assignment #3

by TA Yi-Ting, Lin r07943102@ntu.edu.tw

Due 2 p.m., November 19, 2018

1 Problem Statement

We know that a binary search tree (BST) can be traversed in different ways (i.e., inorder, preorder, and postorder). If a preorder traversal sequence contains no duplicate elements, then the sequence can construct a unique BST. For example, [1,3,2] represents the BST in Figure 1(a) and [2,1,3] represents the BST in Figure 1(b). On the other hand, some sequences cannot represent any legal BSTs. For example, [2,3,1] cannot be a preorder traversal sequence of any BSTs. Besides, postorder traversal sequence has similar property. If a postorder traversal sequence contains no duplicate elements, then the sequence can also construct a unique BST.

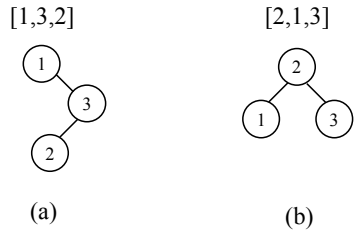

Figure 1: (a) The preorder traversal sequence [1,3,2] and the corresponding BST. (b) The preorder traversal sequence [2,1,3] and the corresponding BST.

This programming assignment asks you to check if a given sequence can be a preorder or a postorder traversal of a BST. If yes, please find the another way to traverse the corresponding BST.

2 Input/Output Specification

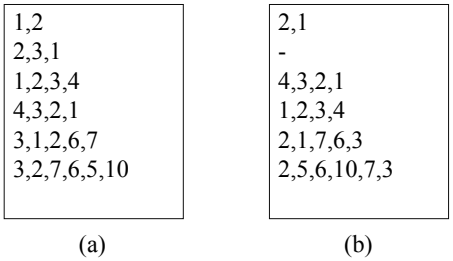

Figure 2: (a) An input example of 6 given preorder traversal sequences. (b) The output example of (a).

2.1 Input Format

The input consists of several sequences separated by a newline. Each input sequence consists several elements (i.e., integers) separated by a comma. Figure 2(a) shows an input example.

2.2 Output Format

The output contains the answers of the input in the order of the input. If the input sequence is a preorder traversal of a binary search tree, the answer should be the postorder traversal of the corresponding BST. Otherwise, the answer should be "-". On the other hand, if we interpret input sequence as a postorder traversal, the answer should be the preorder traversal of the corresponding BST or "-". The output answers should be separated by a newline. Figure 2(b) shows an output example.

3 Command-line Arguments

Please follow the command-line arguments as below. Use "-pre" or "-post" to set how to interpret input sequences, preorder or postorder.
Usage:

python3 programming_hw3.py <-pre|-post> [input file] [output file]

Example:

python3 programming_hw3.py -pre input.txt output.txt

4 Evaluation

To check the correctness of your implementation, 50 public cases are provided for you. In the final evaluation, we will use the 50 public cases and 50 hidden cases (1% for each case) to evaluate your code. The sizes of hidden cases are similar to those of the public cases. The runtime limit for each case is 1 second.

5 Submission

- Please compress your programming_hw3.py (.zip or .tar) and upload it to CEIBA.
- Please submit your code before **2 p.m. on November 19, 2018**.

6 Requirement

- Please use python3 to write the program.
- Do not import any other library. You need to implement the node and tree structure by yourself.