January 2023 CSE 106
Online Assignment on BST
Matching Sorted List

Time: 30 minutes

Subsections B1 & B2

You are given a list of numbers which is guaranteed to be sorted. You need to check if the list is present as a sublist or subsequence in the in-order traversal of the BST.

- **Sublist:** A sublist is a contiguous subsequence of elements taken from a larger list. It is formed by selecting a portion of the original list while maintaining the order of the elements from the original list. The elements in a sublist should be adjacent to each other in the original list. For example, consider the list of numbers: [2, 4, 6, 8, 10, 12]

  Some examples of sublists from this list are:

  [2, 4], [6, 8, 10], [4, 6, 8, 10, 12], [10], [2, 4, 6, 8, 10, 12] (the original list is also a sublist of itself)

- **Subsequence:** A subsequence is a sequence of elements that can be derived from another sequence by deleting some or no elements without changing the order of the remaining elements. Unlike a sublist, a subsequence does not require the elements to be contiguous or adjacent to each other in the original sequence. For example, consider the list of numbers: [1, 2, 3, 4, 5]

  Some examples of subsequences from this list are:

  [1, 3, 5], [2, 4, 5], [1, 2, 4, 5], [1, 2, 3, 4, 5] (the original list is also a subsequence of itself)

This operation is expected to have a time complexity of O(n) where n is the size of the BST.

Please Notice

1. You can use exactly one auxiliary array, stack or queue to store the input list only.

2. Returning the inorder traversal list and manipulating it : not accepted.

Input

A new input command would be check (C) followed by the length of the input list and then the list itself.

Output

For the C command, you need to output

- "sublist" if it is a sublist

- "subsequence" else if it subsequence

- "none" otherwise.

Exact match with the output is expected for a faster evaluation.

Mark Distribution

Task 1 (50%) Correctly output sublist.

Task 2 (30%) Correctly output subsequence.

Task 3 (20%) Correctly output none.

Sample I/O

Input File

```
F 1
I 8
I 10
I 3
C 1 4
C 1 3
I 1
I 14
I 6
I 4
I 13
I 7
C 2 3 4
C 2 3 6
C 3 1 4 7
C 3 1 3 4
C 3 10 13 14
C 3 10 13 15
T In
T Pre
T Post
D 8
D 7
D 10
D 10
C 3 10 13 14
C 3 6 13 14
C 2 6 14
F 4
```

Output

```
not found
(8)
(8(,10))
(8(3,10))
none
sublist
(8(3(1,),10))
(8(3(1,),10(,14)))
(8(3(1,6),10(,14)))
(8(3(1,6(4,)),10(,14)))
(8(3(1,6(4,)),10(,14(13,))))
(8(3(1,6(4,7)),10(,14(13,))))
sublist
subsequence
subsequence
sublist
sublist
none
1 3 4 6 7 8 10 13 14
8 3 1 6 4 7 10 14 13
1 4 7 6 3 13 14 10 8
(10(3(1,6(4,7)),14(13,)))
(10(3(1,6(4,)),14(13,)))
(13(3(1,6(4,)),14))
```

```
(13(3(1,6(4,)),14))
none
sublist
subsequence
found
```

Hints

- Have you noticed inorder traversal always results in a sorted sequence?

- Use a queue to hold the input list. Pass the queue to in-order traversal. If a match is found, dequeue and update the necessary flags. Do it recursively.

- A naive solution for subsequence, call search for each element of the input list.

Submission Guideline

1. Create a directory with your 7 digit student id as its name

2. Put the source files only into the directory created in step 1

3. Zip the directory (compress in .zip format; .rar, .7z or any other format is not acceptable)

4. Upload the .zip file on moodle.

For example, if your student id is 215xxx, create a directory named 2105xxx. Put only your source files (.c, .cpp, .java, .h, etc.) into 215xxx. Compress 215xxx into 215xxx.zip and upload the 215xxx.zip on moodle.