

ASSIGNMENT-4

M. SAI CHARAN
VU21CSEN0300107
CSE-AIML

Unique Binary Search Trees

The screenshot shows the LeetCode problem page for "96. Unique Binary Search Trees". The problem description states: "Given an integer n , return the number of structurally unique BST's (binary search trees) which has exactly n nodes of unique values from 1 to n ." Example 1 shows $n = 3$ and Output: 5. Example 2 shows $n = 1$ and Output: 1. The constraints are $1 \leq n \leq 19$. The code editor shows a Java solution using dynamic programming:

```
1 public class Solution {
2     public int numTrees(int n) {
3         int[] dp = new int[n + 1];
4         dp[0] = 1;
5         for (int i = 1; i <= n; i++) {
6             for (int j = 1; j <= i; j++) {
7                 dp[i] += dp[j - 1] * dp[i - j];
8             }
9         }
10        return dp[n];
11    }
12 }
```

The test result shows "Accepted" with a runtime of 0 ms. The input is $n = 3$ and the output is 5.

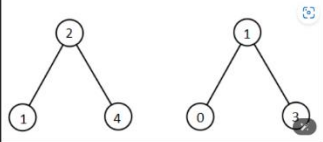
The screenshot shows the LeetCode submission page for "96. Unique Binary Search Trees". The submission is by "MandanaSacharan" and is marked as "Accepted". The runtime is 0 ms, beats 100.00%, and the memory is 40.38 MB, beats 64.32%. The code editor shows the same Java solution as in the previous screenshot. The test result shows "Accepted" with a runtime of 0 ms. The input is $n = 1$ and the output is 1.

All Elements in Two Binary Search Trees

1305. All Elements in Two Binary Search Trees

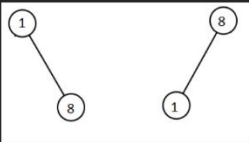
Given two binary search trees `root1` and `root2`, return a list containing all the integers from both trees sorted in **ascending order**.

Example 1:



Input: `root1 = [2,1,4]`, `root2 = [1,0,3]`
Output: `[0,1,1,2,3,4]`

Example 2:



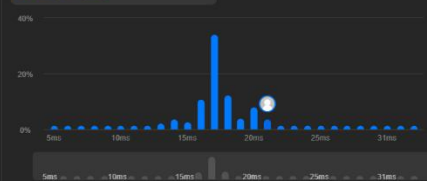
3.1K 19 20 Online

```
Java
11 *   this.left = left;
12 *   this.right = right;
13 * }
14 * }
15 */
16 class Solution {
17     public List<Integer> getAllElements(TreeNode root1, TreeNode root2) {
18         Stack<TreeNode> st1 = new Stack<>();
19         Stack<TreeNode> st2 = new Stack<>();
20
21         List<Integer> res = new ArrayList<>();
22
23         while (root1 != null || root2 != null || !st1.empty() || !st2.empty()) {
24             while (root1 != null) {
25                 st1.push(root1);
26                 root1 = root1.left;
27             }
28             while (root2 != null) {
29                 st2.push(root2);
30                 root2 = root2.left;
31             }
32             if (st2.empty() || (!st1.empty() && st1.peek().val <= st2.peek().val)) {
33                 root1 = st1.pop();
34                 res.add(root1.val);
35                 root1 = root1.right;
36             } else {
37                 root2 = st2.pop();
38                 res.add(root2.val);
39                 root2 = root2.right;
40             }
41         }
42         return res;
43     }
44 }
```

Accepted 48 / 48 testcases passed

MandanaGacharan submitted at Mar 12, 2025 10:41

Runtime: 21 ms, Beats: 17.36%
Memory: 46.82 MB, Beats: 13.00%



Code: Java

```
/**
 * Definition for a binary tree node.
 * public class TreeNode {
 *     int val;
 *     TreeNode left;
 *     TreeNode right;
 *     TreeNode() {}
 *     TreeNode(int val) { this.val = val; }
 * }
 */
```

Testcase: Accepted Runtime: 0 ms

Case 1 Case 2

Input

`root1 = [2,1,4]`

`root2 = [1,0,3]`

Output

`[0,1,1,2,3,4]`

Expected

`[0,1,1,2,3,4]`

Contribute a testcase

Unique Binary Search Trees - Leet...All Elements in Two Binary Search...

https://leetcode.com/problems/all-elements-in-two-binary-search-trees/

Import favoritesGitHub - Colt/TheW...Make your holiday...JobBoard - Preplnst...Profile - JPMC Cand...What Is Web Applic...@jobssforum.in | Link...Other favorites

Problem List<>>RunSubmit

DescriptionEditorialSolutionsAcceptedSubmissions

All Submissions


Accepted48 / 48 testcases passedMandanaSacharan submitted at Mar 12, 2025 10:41

EditorialSolution

Runtime21 msBeats: 17.36%

Memory46.82 MBBeats: 13.00%

Analyze Complexity



CodeJava

```
/**
 * Definition for a binary tree node.
 * public class TreeNode {
 *     int val;
 *     TreeNode left;
 *     TreeNode right;
 *     TreeNode() {}
 *     TreeNode(int val) { this.val = val; }
 * }
```

View more

More challenges

Code

TestcaseTest Result

AcceptedRuntime: 0 ms

Case 1Case 2

Input

root1 =
[1,null,8]

root2 =
[8,1]

Output

[1,1,8,8]

Expected

[1,1,8,8]

Contribute a testcase

28°C
Haze

Search

ENG
IN

10:42:16
12-03-2025