AlgoExpert Quad Layout Java 12px Sublime Monokai 00:00:00

Prompt Scratchpad Our Solution(s) Video Explanation Run Code

Solution 1 Solution 2 Solution 3

```java
// Copyright © 2020 AlgoExpert, LLC. All rights reserved.

import java.util.*;

class Program {
  // O(n^2) time | O(n) space
  public static int numberOfBinaryTreeTopologies(int n) {
    Map<Integer, Integer> cache = new HashMap<Integer, Integer>();
    cache.put(0, 1);
    return numberOfBinaryTreeTopologies(n, cache);
  }

  public static int numberOfBinaryTreeTopologies(int n, Map<Integer, Integer> cache) {
    if (cache.containsKey(n)) {
      return cache.get(n);
    }
    int numberOfTrees = 0;
    for (int leftTreeSize = 0; leftTreeSize < n; leftTreeSize++) {
      int rightTreeSize = n - 1 - leftTreeSize;
      int numberOfLeftTrees = numberOfBinaryTreeTopologies(leftTreeSize, cache);
      int numberOfRightTrees = numberOfBinaryTreeTopologies(rightTreeSize, cache);
      numberOfTrees += numberOfLeftTrees * numberOfRightTrees;
    }
    cache.put(n, numberOfTrees);
    return numberOfTrees;
  }
}

```