

Our Solution(s)

Run Code

Your Solutions

Run Code

Solution 1Solution 2Solution 3

```
1 // Copyright © 2020 AlgoExpert, LLC. All rights reserved.
2
3 import java.util.*;
4
5 class Program {
6     // O(n^2) time | O(n) space
7     public static int numberOfBinaryTreeTopologies(int n) {
8         Map<Integer, Integer> cache = new HashMap<Integer, Integer>();
9         cache.put(0, 1);
10        return numberOfBinaryTreeTopologies(n, cache);
11    }
12
13    public static int numberOfBinaryTreeTopologies(int n, Map<Integer, Integer> cache) {
14        if (cache.containsKey(n)) {
15            return cache.get(n);
16        }
17        int numberOfTrees = 0;
18        for (int leftTreeSize = 0; leftTreeSize < n; leftTreeSize++) {
19            int rightTreeSize = n - 1 - leftTreeSize;
20            int numberOfLeftTrees = numberOfBinaryTreeTopologies(leftTreeSize, cache);
21            int numberOfRightTrees = numberOfBinaryTreeTopologies(rightTreeSize, cache);
22            numberOfTrees += numberOfLeftTrees * numberOfRightTrees;
23        }
24        cache.put(n, numberOfTrees);
25        return numberOfTrees;
26    }
27 }
28
```

Solution 1Solution 2Solution 3

```
1 class Program {
2     public static int numberOfBinaryTreeTopologies(int n) {
3         // Write your code here.
4         return -1;
5     }
6 }
7
```

```
10 # Write event to test log as value of the test log message
11 }
12
13 # Test
14 # Write event to test log as value of the test log message
15 # Write event to test log as value of the test log message
16 }
17
18 # Test
19 # Write event to test log as value of the test log message
20 }
21
22 # Test
```

Run or submit code when you're ready.