



**Output #00:**

5

**Explanation:**

The diameter of the tree is 5

**YOUR ANSWER**

Draft saved 05:23 pm

Java



```
1 ▼ import java.io.*;
2  import java.util.*;
3  import java.text.*;
4  import java.math.*;
5  import java.util.regex.*;
6
7  public class Solution {
8
9  ▼ private static class Node {
10      Node left, right;
11      int data;
12  ▼   Node(int newData) {
13          left = right = null;
14          data = newData;
15      }
16  }
17
18  ▼ private static Node insert(Node node, int data) {
19  ▼      if (node==null){
20          node = new Node(data);
21      }
22  ▼      else {
23  ▼          if (data < node.data) {
24              node.left = insert(node.left, data);
```

```
25         }
26     ▼     else {
27         node.right = insert(node.right, data);
28     }
29     }
30     return(node);
31 }
32
33 ▼ private static Node addRandomElement(Node node, int data,
    int index){
34     ▼     if (node==null){
35         node = new Node(data);
36     }
37     ▼     else {
38     ▼         if (index <= 2) {
39             node.left = addRandomElement(node.left,
data,index);
40         }
41     ▼         else {
42             node.right = addRandomElement(node.right,
data,index);
43         }
44     }
45     }
46     return(node);
47 }
48
49 ▼ static int diameterOfTree(Node root) {
50     return diameterOfTreeHelper(root).diameter;
51 }
52
53 ▼ static Pair diameterOfTreeHelper(Node root) {
54     ▼ if (root == null) {
55         return new Pair(0,0);
56     }
57     Node L = root.left;
58     Node R = root.right;
59     Pair LP = diameterOfTreeHelper(L);
60     Pair RP = diameterOfTreeHelper(R);
61     int height = Math.max(LP.height+1, RP.height+1);
62     int candidate3 = LP.height + RP.height + 1;
63     int diameter = Math.max(candidate3,
Math.max(LP.diameter, RP.diameter));
64     return new Pair(height, diameter);
65 }
66 }
```

```
66
67 ▼ private static class Pair {
68     int height, diameter;
69 ▼ Pair(int h, int d) {
70     height = h;
71     diameter = d;
72 }
73 }
74 ▼ public static void main(String[] args) throws IOException
    {
```



Sample Test

⌚ 00:18:43  
to test end

1

2

3

4

```
78
79     Node _root;
80     int root_i=0, root_cnt = 0, root_num = 0;
81     _root = null;
82     int isBst = in.nextInt();
83     root_cnt = in.nextInt();
84 ▼ for(root_i = 0; root_i < root_cnt; root_i++){
85     root_num = in.nextInt();
86     if( isBst == 0 ){                                _root =
addRandomElement(_root,root_num,root_i);
87 ▼     } else {
88         _root = insert(_root, root_num);
89     }
90 }
91
92     bw.write(String.valueOf(diameterOfTree(_root)));
93     bw.newLine();
94     bw.close();
95     return;
96 }
97
98 }
```

Line: 71 Col: 18

☐ Test against custom input

Run Code

Submit code &amp; Continue

⬇ Download sample testcases    The input/output files have Unix line endings.  
Do not use Notepad to edit them on windows.

**Status: Compiled successfully. All available test cases passed!**

---

**Testcase 1: Success**

**Your Output**

Output hidden

---

**Testcase 2: Success**

**Your Output**

5

**Expected Output**

5

---

**Testcase 3: Success**

**Your Output**

Output hidden

---

**Testcase 4: Success**

**Your Output**

Output hidden

---

**Testcase 5: Success**

**Your Output**

Output hidden

---

**Testcase 6: Success**

**Your Output**

Output hidden

---

### Testcase 7: Success

Your Output

Output hidden

---

### Testcase 8: Success

Your Output

Output hidden

---

### Testcase 9: Success

Your Output

Output hidden

---

### Testcase 10: Success

Your Output

Output hidden

---

### Testcase 11: Success

Your Output

Output hidden

---

### Testcase 12: Success

Your Output

Output hidden

---

### Testcase 13: Success

Your Output

Output hidden

### Testcase 14: Success

Your Output

Output hidden

### Testcase 15: Success

Your Output

Output hidden

[About](#) [Privacy policy](#) [Terms of service](#)