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

Scratchpad Our Solution(s) Video Explanation Run Code

```
Solution 1 Solution 2
```

Prompt

40

```
1\, // Copyright @ 2020 AlgoExpert, LLC. All rights reserved.
 3 #include <vector>
 4 using namespace std;
 6 class BinaryTree {
 7 public:
    int value;
     BinaryTree *left = NULL;
     BinaryTree *right = NULL;
10
12
     BinaryTree(int value);
13 };
14
15 vector<BinaryTree *> getNodesInOrder(BinaryTree *tree,
16
                                    vector<BinaryTree *> *array);
17
19
   BinaryTree *flattenBinaryTree(BinaryTree *root) {
     vector<BinaryTree *> inOrderNodes =
20
        getNodesInOrder(root, new vector<BinaryTree *>{});
21
22
     for (int i = 0; i < inOrderNodes.size() - 1; i++) {</pre>
      BinaryTree *leftNode = inOrderNodes[i];
23
24
       BinaryTree *rightNode = inOrderNodes[i + 1];
25
       leftNode->right = rightNode;
26
       rightNode->left = leftNode;
27
     return inOrderNodes[0];
28
29 }
30
31
   vector<BinaryTree *> getNodesInOrder(BinaryTree *tree,
32
                                    vector<BinaryTree *> *array) {
33
     if (tree != NULL) {
34
       getNodesInOrder(tree->left, array);
35
       array->push_back(tree);
36
       getNodesInOrder(tree->right, array);
37
38
     return *array;
39 }
```