

Our Solution(s)

Run Code

Solution 1Solution 2

1 // Copyright © 2020 AlgoExpert, LLC. All rights reserved.

2

3 class Program {

4 // O(n) time | O(d) space

5 func invertBinaryTree(tree: BinaryTree?) {

6 if let tree = tree {

7 swapLeftAndright(tree: tree)

8 invertBinaryTree(tree: tree.left)

9 invertBinaryTree(tree: tree.right)

10 }

11 }

12

13 func swapLeftAndright(tree: BinaryTree) {

14 let left = tree.left

15 tree.left = tree.right

16 tree.right = left

17 }

18 }

19

Our Tests

Your Solutions

Run Code

Solution 1Solution 2Solution 3

1 class Program {

2 func invertBinaryTree(tree: BinaryTree) {

3 // Write your code here.

4 }

5 }

6

Custom Output

Submit Code

```

1  def is_palindrome(word):
2      if word == word[::-1]:
3          return True
4      else:
5          return False
6
7  # Test the function with "level" and "racecar"
8  print(is_palindrome("level"))
9  print(is_palindrome("racecar"))
10
11 # Test the function with "hello" and "world"
12 print(is_palindrome("hello"))
13 print(is_palindrome("world"))
14
15 # Test the function with "a" and "12345"
16 print(is_palindrome("a"))
17 print(is_palindrome("12345"))
18
19 # Test the function with "12321" and "54321"
20 print(is_palindrome("12321"))
21 print(is_palindrome("54321"))
22

```

Run or submit code when you're ready.