

PromptScratchpadOur Solution(s)Video ExplanationRun Code

Solution 1Solution 2

```
1 // Copyright © 2020 AlgoExpert, LLC. All rights reserved.
2
3 class Program {
4     // O(n * n!) time | O(n * n!) space
5     func permutations(array: inout [Int]) -> [[Int]] {
6         var permutations = [[Int]]()
7         permutationsHelper(firstIndex: 0, array: &array, permutations: &permutations)
8
9         return permutations
10    }
11
12    func permutationsHelper(firstIndex: Int, array: inout [Int], permutations: inout [[Int]]
13        if firstIndex == array.count - 1 {
14            permutations.append(array)
15        } else {
16            for secondIndex in firstIndex ..< array.count {
17                swap(firstIndex: firstIndex, secondIndex: secondIndex, array: &array)
18                permutationsHelper(firstIndex: firstIndex + 1, array: &array, permutations:
19                    swap(firstIndex: firstIndex, secondIndex: secondIndex, array: &array)
20            }
21        }
22    }
23
24    func swap(firstIndex: Int, secondIndex: Int, array: inout [Int]) {
25        let temp = array[firstIndex]
26        array[firstIndex] = array[secondIndex]
27        array[secondIndex] = temp
28    }
29 }
30
```

Your SolutionsRun Code

Solution 1Solution 2Solution 3

```
1 class Program {
2     func permutations(array: inout [Int]) -> [[Int]] {
3         // Write your code here.
4         return []
5     }
6 }
7
```

Run or submit code when you're ready.