Our Solution(s)

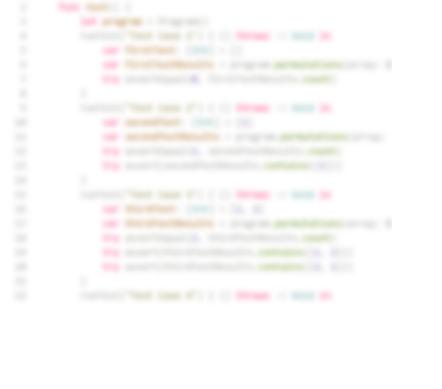
Run Code

```
Your Solutions Run Code
```

Sublime

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

```
class Program {
   func permutations(array: inout [Int]) -> [[Int]] {
      // Write your code here.
      return []
}
```



Run or submit code when you're ready.