

Our Solution(s)

Run Code

Your Solutions

Run Code

Solution 1

Solution 2

```
1 // Copyright © 2020 AlgoExpert, LLC. All rights reserved.
2
3 package main
4
5 // Upper Bound: O(n^2*n!) time | O(n*n!) space
6 // Roughly: O(n*n!) time | O(n*n!) space
7 func GetPermutations(array []int) [][]int {
8     permutations := [][]int{}
9     permutationsHelper(array, []int{}, &permutations)
10    return permutations
11 }
12
13 func permutationsHelper(array []int, currentPermutation []int, permutations *[][]int) {
14     if len(array) == 0 && len(currentPermutation) != 0 {
15         *permutations = append(*permutations, currentPermutation)
16         return
17     }
18     for i := range array {
19         newArray := make([]int, i)
20         copy(newArray, array[:i])
21         newArray = append(newArray, array[i+1:]...)
22         newPermutation := make([]int, len(currentPermutation))
23         copy(newPermutation, currentPermutation)
24         newPermutation = append(newPermutation, array[i])
25         permutationsHelper(newArray, newPermutation, permutations)
26     }
27 }
28
```

Solution 1

Solution 2

Solution 3

```
1 package main
2
3 func GetPermutations(array []int) [][]int {
4     // Write your code here.
5     return nil
6 }
7
```

Custom Output

Raw Output

Submit Code

Run or submit code when you're ready.