Our Solution(s)

Run Code

Solution 1   **Solution 2**

```go
// Copyright © 2020 AlgoExpert, LLC. All rights reserved.

package main

// O(n*n!) time | O(n*n!) space
func GetPermutations(array []int) [][]int {
  permutations := [][]int{}
  permutationsHelper(0, array, &permutations)
  return permutations
}

func permutationsHelper(i int, array []int, permutations *[][]int) {
  if i == len(array)-1 {
    newPerm := make([]int, len(array))
    copy(newPerm, array)
    *permutations = append(*permutations, newPerm)
    return
  }
  for j := i; j < len(array); j++ {
    swap(array, i, j)
    permutationsHelper(i+1, array, permutations)
    swap(array, i, j)
  }
}

func swap(array []int, i, j int) {
  array[i], array[j] = array[j], array[i]
}
```

Your Solutions

Run Code

Solution 1   **Solution 2**   **Solution 3**

```go
package main

func GetPermutations(array []int) [][]int {
  // Write your code here.
  return nil
}
```

Custom Output   **Raw Output**

Submit Code

Run or submit code when you're ready.