Our Solution(s)   Run Code

Your Solutions   Run Code

Solution 1   Solution 2   Solution 1   Solution 2   Solution 3

```python
# Copyright © 2020 AlgoExpert, LLC. All rights reserved.

# Upper Bound: O(n^2*n!) time | O(n*n!) space
# Roughly: O(n*n!) time | O(n*n!) space
def getPermutations(array):
    permutations = []
    permutationsHelper(array, [], permutations)
    return permutations


def permutationsHelper(array, currentPermutation, permutations):
    if not len(array) and len(currentPermutation):
        permutations.append(currentPermutation)
    else:
        for i in range(len(array)):
            newArray = array[:i] + array[i + 1 :]
            newPermutation = currentPermutation + [array[i]]
            permutationsHelper(newArray, newPermutation, permutations)
```

```python
def getPermutations(array):
    # Write your code here.
    pass
```

Our Solution(s)   Run Code

Your Solutions

Custom Output   Raw Output   Submit Code

```python
# Copyright © 2020 AlgoExpert, LLC. All rights reserved.

# Upper Bound: O(n^2*n!) time | O(n*n!) space
# Roughly: O(n*n!) time | O(n*n!) space
def getPermutations(array):
    permutations = []
    return permutations
```

```python
def getPermutations(array):
    # Write your code here.
    pass
```

Run or submit code when you're ready.