Permutation of Array [LeetCode](https://leetcode.com/problems/permutations/)

This C++ program generates all permutations of a given array using a backtracking approach.

**Approach 1: Function to find all permutations of the array using Backtracking**

* + The **findPermutations** function generates all permutations of the input array **nums** using recursive backtracking.
  + It iterates through the array starting from the given **index**.
  + At each step, it swaps the element at the current index with the element at index **i**, where **i** varies from the current index to the end of the array.
  + After swapping, it recursively explores permutations for the next index.
  + Once the recursive call returns, it backtracks by swapping the elements back to their original positions.
  + When the **index** becomes equal to or greater than the array size, the current permutation is added to the answer.
  + **Time Complexity: O(n!), where n is the length of the input array nums. There are n! permutations to generate.**
  + **Space Complexity: O(n), as the maximum depth of the recursive call stack is n.**

**Permutation Function (permutation):**

* + The **permutation** function initializes the **ans** vector and starts the backtracking process by calling the **findPermutations** function with the initial index as 0.
  + It returns the generated permutations.