* Find all permutations of nums

  * nums: [1,2,3]

```cpp
void DFS(vector<vector<int>> &ans, vector<int>& nums, int start) {
    // DFS in this problem is not intuitive, it's better to see the graph in github repo.
    // TC of this function itselt: O(N)
    // start is the pivot index which nums[start]
    if (start == nums.size()) {
        ans.push_back(nums);
        return;
    }
    for (size_t i = start; i < nums.size(); i++) {
        swap(nums[start], nums[i]); // original permutation
        DFS(ans, nums, start + 1);  // When all permutations have been exhausted,
        // return to the current scope and begin backtracking through other permutations.
        swap(nums[start], nums[i]); // resumes to original permutation
    }
}
```