46. Permutations

Given an array nums of distinct integers, return all the possible permutations. You can return the answer in any order.

Example 1:
Input: nums = [1,2,3]
Output: [[1,2,3],[1,3,2],[2,1,3],[2,3,1],[3,1,2],[3,2,1]]

Example 2:
Input: nums = [0,1]
Output: [[0,1],[1,0]]

Example 3:
Input: nums = [1]
Output: [[1]]

Constraints:

- 1 <= nums.length <= 6
- -10 <= nums[i] <= 10
- All the integers of nums are unique.
-

131. Palindrome Partitioning

Given a string s, partition s such that every
substring
of the partition is a
palindrome
. Return all possible palindrome partitioning of s.

Example 1:
Input: s = "aab"
Output: [["a","a","b"],["aa","b"]]

Example 2:
Input: s = "a"
Output: [["a"]]

Constraints:

- 1 <= s.length <= 16
- s contains only lowercase English letters.

90. Subsets II

Given an integer array nums that may contain duplicates, return all possible subsets
(the power set).
The solution set must not contain duplicate subsets. Return the solution in any order.

Example 1:
Input: nums = [1,2,2]
Output: [[],[1],[1,2],[1,2,2],[2],[2,2]]

Example 2:
Input: nums = [0]
Output: [[],[0]]

Constraints:

- 1 <= nums.length <= 10
- -10 <= nums[i] <= 10

39. Combination Sum

Given an array of distinct integers candidates and a target integer target, return a list of all unique combinations of candidates where the chosen numbers sum to target. You may return the combinations in any order.
The same number may be chosen from candidates an unlimited number of times. Two combinations are unique if the
frequency

of at least one of the chosen numbers is different.
The test cases are generated such that the number of unique combinations that sum up to target is less than 150 combinations for the given input.

Example 1:
Input: candidates = [2,3,6,7], target = 7
Output: [[2,2,3],[7]]
Explanation:
2 and 3 are candidates, and 2 + 2 + 3 = 7. Note that 2 can be used multiple times.
7 is a candidate, and 7 = 7.
These are the only two combinations.

Example 2:
Input: candidates = [2,3,5], target = 8
Output: [[2,2,2,2],[2,3,3],[3,5]]

Example 3:
Input: candidates = [2], target = 1
Output: []

Constraints:

- 1 <= candidates.length <= 30
- 2 <= candidates[i] <= 40
- All elements of candidates are distinct.
- 1 <= target <= 40

40. Combination Sum II

Given a collection of candidate numbers (candidates) and a target number (target), find all unique combinations in candidates where the candidate numbers sum to target.
Each number in candidates may only be used once in the combination.
Note: The solution set must not contain duplicate combinations.

Example 1:

Input: candidates = [10,1,2,7,6,1,5], target = 8
Output:
[
[1,1,6],
[1,2,5],
[1,7],
[2,6]
]

Example 2:
Input: candidates = [2,5,2,1,2], target = 5
Output:
[
[1,2,2],
[5]
]

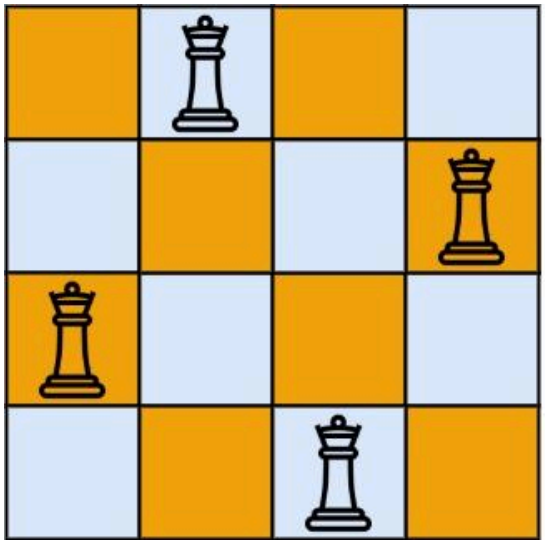
Constraints:

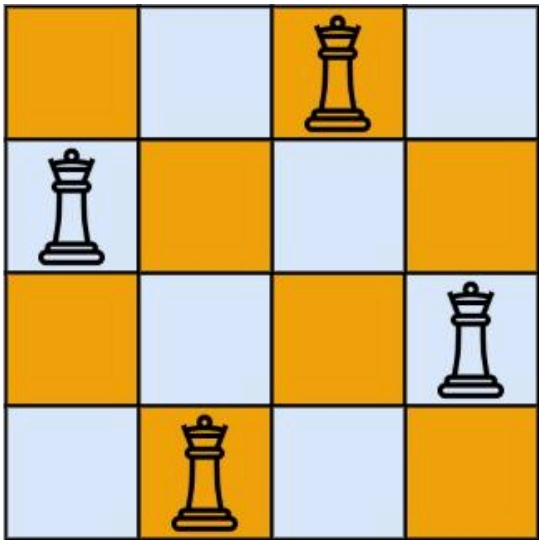
- 1 <= candidates.length <= 100
- 1 <= candidates[i] <= 50
- 1 <= target <= 30

51. N-Queens

The n-queens puzzle is the problem of placing n queens on an n x n chessboard such that no two queens attack each other.
Given an integer n, return all distinct solutions to the n-queens puzzle. You may return the answer in any order.
Each solution contains a distinct board configuration of the n-queens' placement, where 'Q' and '.' both indicate a queen and an empty space, respectively.

Example 1:





Input: n = 4
Output: [[".Q..","...Q","Q...","..Q."],["..Q.","Q...","...Q",".Q.."]]
Explanation: There exist two distinct solutions to the 4-queens puzzle as shown above

Example 2:
Input: n = 1
Output: [["Q"]]

Constraints:

- 1 <= n <= 9