

# Leetcode Hard Questions

65

Valid number

Communicate to get what the interviewer really wants

149

Max Points on a line

for for loop.  $O(n^2)$  time complexity, sort the points according to X, count all other points with the same slope, same points, and no slope. use a map to record these 3 cases, then find the max of these 3 conditions.

126

Word Ladder II

BFS to find the minimum steps. Early stopping DFS to find all the ways.

862

Shortest Subarray with Sum at Least K

Presum the array, Use Deque to maintain an increasing order of the presum. Only increasing presum would give possible result. Then check every time from the first to the current index.

564

Find the Closest Palindrome

\*\*\*\*\*

Annoying Math

420

Strong Password Checker

\*\*\*\*\*

68

Text Justification

Simulate the process, need to look forward or look back. Be careful.

146

LRU Cache

HashMap + doubly linked list

805

Split Array With Same Average

\*\*\*\*\*

related to 416. Partition Equal Subset Sum. This is a 0/1 knapsack problem. Use the 2D array for easy understanding.

```
class Solution {
    public boolean canPartition(int[] nums) {
        if (nums == null || nums.length == 0) {
            return true;
        }

        int sum = 0;
        for (int n : nums) {
            sum += n;
        }
        if (sum % 2 != 0) {
            return false;
        }

        sum /= 2;
        boolean[][] dp = new boolean[nums.length + 1][sum +
1];

        for (int i = 0; i < nums.length + 1; i++) {
            dp[i][0] = true;
        }

        for (int i = 1; i <= nums.length; i++) {
```

```
        for (int j = 1; j <= sum; j++) {
            dp[i][j] = dp[i - 1][j];
            if (j - nums[i - 1] >= 0) {
                dp[i][j] = (dp[i][j] || dp[i - 1][j - nums
[i - 1]]);
            }
        }
    }
    return dp[nums.length][sum];
}
```