Greedy Algorithm Confidence Builder (HackWithInfy Focused)

Solve all problems using Greedy Algorithm as the primary strategy in the sequence below to build your confidence from easy to hard levels.

Warm-Up (Easy Level)

Problem 1: Buy Two Chocolates

LeetCode: 2706. Buy Two Chocolates
YouTube Solution: Buy Two Chocolates | Code Decode

Problem Description:

You are given an array of prices of chocolates. Find the minimum total cost of buying any two chocolates. If you can't buy two chocolates, return the amount you have.

Input Format:

- n → Number of chocolates
- prices[] → Array of prices
- money → Amount you have

Output Format:

- Remaining money after buying the two cheapest chocolates.

Constraints:

- $2 \leq n \leq 100$
- $1 \leq prices[i] \leq 100$
- $1 \leq money \leq 100$

Sample Input:

5
1 2 2 5 7
10

Sample Output:

7

Problem 2: Array Partition

LeetCode: 561. Array Partition
YouTube Solution: Array Partition | NeetCode

Problem Description:

Given an array of 2n integers, group these integers into n pairs to maximize the sum of the minimums in each pair.

Input Format:

- n → Number of pairs
- nums[] → Array of 2n integers

Output Format:

- Maximum sum of minimums in all pairs.

Constraints:

- $1 \leq n \leq 10000$
- $-10000 \leq nums[i] \leq 10000$

Sample Input:

4
1 4 3 2

Sample Output:

4

Problem 3: DI String Match

LeetCode: 942. DI String Match
YouTube Solution: DI String Match | NeetCode

Problem Description:

You are given a string s consisting of letters 'D' and 'I'. Reconstruct a permutation of the integers [0, 1, ..., n] that matches the pattern 'D' = decreasing, 'I' = increasing.

Input Format:

- s → A string of length n

Output Format:

- Array representing the permutation

Constraints:

- $1 \leq s.length \leq 1000$

Sample Input:

IDID

Sample Output:

[0, 4, 1, 3, 2]

Skill Booster (Medium Level)

Problem 4: Jump Game

LeetCode: 55. Jump Game
YouTube Solution: Jump Game | NeetCode

Problem Description:

You are given an array where each element represents your maximum jump length at that position. Determine if you can reach the last index.

Input Format:

- n → Number of elements
- nums[] → Array of jump lengths

Output Format:

- Print true if reachable, otherwise false.

Constraints:

- $1 \leq n \leq 10^4$
- $0 \leq nums[i] \leq 10^5$

Sample Input:

5
2 3 1 1 4

Sample Output:

true

Problem 5: Jump Game II

LeetCode: 45. Jump Game II
YouTube Solution: Jump Game II | NeetCode

Problem Description:

You are given an array where each element represents your maximum jump length. Find the minimum number of jumps required to reach the last index.

Input Format:

- n → Number of elements
- nums[] → Array of jump lengths

Output Format:

- Minimum number of jumps.

Constraints:

- 1 ≤ n ≤ 10^4
- 0 ≤ nums[i] ≤ 1000

Sample Input:

5
2 3 1 1 4

Sample Output:

2

Problem 6: Best Time to Buy and Sell Stock with Transaction Fee

LeetCode: 714. Best Time to Buy and Sell Stock with Transaction Fee
YouTube Solution: Best Time to Buy and Sell Stock with Transaction Fee | NeetCode

Problem Description:

You are given an array of prices where prices[i] is the price of a stock on day i. You may complete as many transactions as you like, but pay a transaction fee for each sale. Find the maximum profit.

Input Format:

- n → Number of days
- prices[] → Array of stock prices
- fee → Transaction fee

Output Format:

- Maximum profit.

Constraints:

- 1 ≤ n ≤ 5 * 10^4

- $0 \leq prices[i] \leq 50,000$
- $0 \leq fee \leq 500$

Sample Input:

6
1 3 2 8 4 9
2

Sample Output:

8

Challenge Zone (Hard Level)

Problem 7: Reducing Dishes

LeetCode: 1402. Reducing Dishes
YouTube Solution: Reducing Dishes | NeetCode

Problem Description:

You are given an array of satisfaction levels. You can cook dishes in any order. The time coefficient increases by 1 for each dish cooked. Maximize the total satisfaction.

Input Format:

- n → Number of dishes
- satisfaction[] → Array of satisfaction levels

Output Format:

- Maximum total satisfaction.

Constraints:

- $1 \leq n \leq 500$
- $-1000 \leq satisfaction[i] \leq 1000$

Sample Input:

4
-1 -8 0 5

Sample Output:

14

Problem 8: Maximum Spending After Buying Items

LeetCode: 2931. Maximum Spending After Buying Items
Solve independently: No available YouTube solution.

Problem Description:

You are given a matrix where each row represents a store's items from cheapest to costliest. Each day you can buy the cheapest available item across all stores. Maximize the total spending by selecting items wisely.

Input Format:

- m → Number of stores
- n → Number of items per store
- grid[m][n] → Prices of items in each store

Output Format:

- Maximum total spending.

Constraints:

- $1 \leq m, n \leq 50$
- $1 \leq grid[i][j] \leq 100$

Sample Input:

3 3
1 2 3
3 2 1
4 4 4

Sample Output:

24

Summary Table

Problem	Level	LeetCode Link	YouTube Link
Buy Two Chocolates	Easy	2706	Video
Array Partition	Easy	561	Video
DI String Match	Easy	942	Video
Jump Game	Medium	55	Video
Jump Game II	Medium	45	Video
Best Time to Buy and Sell Stock with Fee	Medium	714	Video
Reducing Dishes	Hard	1402	Video
Maximum Spending After Buying Items	Hard	2931	Solve Independently