

Greedy Algorithms (Assignment Questions)

Question 1 : Split a String in Balanced Strings

Balanced strings are those that have an equal quantity of 'L' and 'R' characters.

Given a balanced string *s*, split it into some number of substrings such that: Each substring is balanced.

Return the maximum number of balanced strings you can obtain. [[Go to Qs](#)]

Examples :

Input: s = "RLRLLRLRL"

Output: 4

Explanation: s can be split into "RL", "RLL", "RL", "RL", each substring contains the same number of 'L' and 'R'.

Question 2 : Largest Odd Number in String

You are given a string *num*, representing a large integer. Return the largest-valued odd integer (as a string) that is a non-empty substring of *num*, or an empty string "" if no odd integer exists.

A substring is a contiguous sequence of characters within a string. [[Go to Qs](#)]

Examples :

Input: num = "52"

Output: "5"

Explanation: The only non-empty substrings are "5", "2", and "52". "5" is the only odd number.

Question 3 : Smallest String With A Given Numeric Value

The numeric value of a lowercase character is defined as its position (1-indexed) in the alphabet, so the numeric value of a is 1, the numeric value of b is 2, the numeric value of c is 3, and so on.

The numeric value of a string consisting of lowercase characters is defined as the sum of its characters' numeric values. For example, the numeric value of the string "abe" is equal to $1 + 2 + 5 = 8$.

You are given two integers n and k . Return the lexicographically smallest string with length equal to n and numeric value equal to k .

Note that a string x is lexicographically smaller than string y if x comes before y in dictionary order, that is, either x is a prefix of y , or if i is the first position such that $x[i] \neq y[i]$, then $x[i]$ comes before $y[i]$ in alphabetic order. [[Go to Qs](#)]

Example :

Input: $n = 3, k = 27$

Output: "aay"

Explanation: The numeric value of the string is $1 + 1 + 25 = 27$, and it is the smallest string with such a value and length equal to 3.

Question 4 : Best Time to Buy and Sell Stock

You are given an array `prices` where `prices[i]` is the price of a given stock on the i th day. You want to maximize your profit by choosing a single day to buy one stock and choosing a different day in the future to sell that stock.

Return the maximum profit you can achieve from this transaction. If you cannot achieve any profit, return 0. [[Go to Qs](#)]

Example :

Input: `prices = [7,1,5,3,6,4]`

Output: 5

Explanation: Buy on day 2 (price = 1) and sell on day 5 (price = 6), profit = $6 - 1 = 5$.

Note that buying on day 2 and selling on day 1 is not allowed because you must buy before you sell.

Note : This question was already covered in class lectures of previous chapters.

Question 5 : Split Array Largest Sum

Given an integer array `nums` and an integer `k`, split `nums` into `k` non-empty subarrays such that the largest sum of any subarray is minimized. Return the minimized largest sum of the split. (A subarray is a contiguous part of the array.) [[Go to Qs](#)]

Example :

Input: `nums = [7,2,5,10,8]`, `k = 2`

Output: 18

Explanation: There are four ways to split `nums` into two subarrays.

The best way is to split it into `[7,2,5]` and `[10,8]`, where the largest sum among the two subarrays is only 18.

