Python Coding Assessment

1. REVERSE THE STRING

Given an input strings, reverse the order of the words.

A word is defined as a sequence of non-space characters. The words in s will be separated by at least one space.

Return a string of the words in reverse order concatenated by a single space.

Note that s may contain leading or trailing spaces or multiple spaces between two words. The returned string should only have a single space separating the words. Do not include any extra spaces.

Example 1:

Input: s = "the sky is blue"
Output: "blue is sky the"
Example 2:

Input: s = "hello world"
Output: "world hello"
Explanation: Your reversed string should not contain leading or trailing spaces.
Example 3:

Input: s = "a good example"
Output: "example good a"
Explanation: You need to reduce multiple spaces between two words to a single space in the reversed string.

Constraints:

1 <= s.length <= 104
s contains English letters (upper-case and lower-case), digits, and spaces ' '.
There is at least one word in s.

2. STRING COMPRESSION
Given an array of characters chars, compress it using the following algorithm:

Begin with an empty string s. For each group of consecutive repeating characters in chars:

If the group's length is 1, append the character to s.
Otherwise, append the character followed by the group's length.
The compressed string s should not be returned separately, but instead, be stored in the input character array chars. Note that group lengths that are 10 or longer will be split into multiple characters in chars.

After you are done modifying the input array, return the new length of the array.

You must write an algorithm that uses only constant extra space.

Example 1:

Input: chars = ["a", "a", "b", "b", "c", "c", "c"]
Output: Return 6, and the first 6 characters of the input array should be: ["a","2","b","2","c","3"]
Explanation: The groups are "aa", "bb", and "ccc". This compresses to "a2b2c3".
Example 2:

Input: chars = ["a"]
Output: Return 1, and the first character of the input array should be: ["a"]
Explanation: The only group is "a", which remains uncompressed since it's a single character.
Example 3:

Input: chars = ["a", "b", "b", "b", "b", "b", "b", "b", "b", "b", "b", "b", "b"]
Output: Return 4, and the first 4 characters of the input array should be: ["a","b","1","2"].
Explanation: The groups are "a" and "bbbbbbbbbbbb". This compresses to "ab12".

Constraints:

1 <= chars.length <= 2000
Chars[i] is a lowercase English letter, uppercase English letter, digit, or symbol.

3. Given Temperatures

Given an array of integer's temperatures represents the daily temperatures, return an array answer such that answer[i] is the number of days you have to wait after the ith day to get a warmer temperature. If there is no future day for which this is possible, keep answer[i] == 0 instead.

Example 1:

Input: temperatures = [73, 74, 75, 71, 69, 72, 76, 73]
Output: [1, 1, 4, 2, 1, 1, 0, 0]
Example 2:

Input: temperatures = [30, 40, 50, 60]
Output: [1, 1, 1, 0]
Example 3:

Input: temperatures = [30, 60, 90]
Output: [1, 1, 0]

Constraints:

1 <= temperatures.length <= 105
30 <= temperatures[i] <= 100

4. Find the closest pair from two sorted arrays
Given two sorted arrays and a number x, find the pair whose sum is closest to x and the pair has an element from each array.

We are given two arrays ar1 [0...m-1] and ar2 [0..n-1] and a number x, we need to find the pair ar1 [i] + ar2 [j] such that absolute value of (ar1 [i] + ar2[j] - x) is minimum.

Example:
Input: ar1] = (1, 4, 5, 7); ar2[]= (10, 20, 30, 40); x = 32

Output: 1 and 30

Input: ar1[] = {1, 4, 5, 7); ar2[] = (10, 20, 30, 40); x = 58

Output: 7 and 40

5. Longest palindromic string

Given a string s, return the longest palindromic substring in s.

Example 1:

Input: s = "babad"
Output: "bab"
Explanation: "aba" is also a valid answer.
Example 2:

Input: s = "cbbd"
Output: "bb"

Constraints:

1 <= s.length <= 1000
s consist of only digits and English letters.

6. Longest substring without repeating character

Given a string s, find the length of the longest substring without repeating characters.
Example 1:

Input: s = "abcabcbb"
Output: 3
Explanation: The answer is "abc", with the length of 3.
Example 2:

Input: s = "bbbbb"
Output: 1
Explanation: The answer is "b", with the length of 1.
Example 3:

Input: s = "pwwkew"
Output: 3
Explanation: The answer is "wke", with the length of 3.
Notice that the answer must be a substring, "pwke" is a subsequence and not a substring.

Constraints:

0 <= s.length <= 5 * 104
s consists of English letters, digits, symbols and spaces.

7. SORT colors

Given an array nums with n objects colored red, white, or blue, sort them in-place so that objects of the same color are adjacent, with the colors in the order red, white, and blue.

We will use the integers 0, 1, and 2 to represent the color red, white, and blue, respectively.

You must solve this problem without using the library's sort function.

Example 1:

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

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

Constraints:

n == nums.length
1 <= n <= 300
nums[i] is either 0, 1, or 2.

8. Removing stars from string
You are given a string s, which contains stars *.

In one operation, you can:

Choose a star in s.
Remove the closest non-star character to its left, as well as remove the star itself.
Return the string after all stars have been removed.

Note:

The input will be generated such that the operation is always possible.
It can be shown that the resulting string will always be unique.

Example 1:

Input: s = "leet**cod*e"
Output: "lecoe"
Explanation: Performing the removals from left to right:
- The closest character to the 1st star is't' in "leet**cod*e". s becomes "lee*cod*e".
- The closest character to the 2nd star is 'e' in "lee*cod*e". s becomes "lecod*e".
- The closest character to the 3rd star is'd' in "lecod*e". s becomes "lecoe".
There are no more stars, so we return "lecoe".
Example 2:

Input: s = "erase*****"
Output: ""
Explanation: The entire string is removed, so we return an empty string.

Constraints:

1 <= s.length <= 105
s consists of lowercase English letters and stars *.
The operation above can be performed on s.

9. Given a sequence of words, print all anagrams together

Given an array of words, print all anagrams together. For example, if the given array is {"cat", "dog", "tac", "god", "act"}, then output may be "cat tac act dog god"

10. 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 ith 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.

Example 1:

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.
Example 2:

Input: prices = [7, 6, 4, 3, 1]
Output: 0
Explanation: In this case, no transactions are done and the max profit = 0.

Constraints:

1 <= prices.length <= 105
0 <= prices[i] <= 104