Facebook Interview Problems

1) Given a string, find out if the string is K-Palindrome or not. A K-palindrome string transforms into a palindrome on removing at most k characters from it.

Examples:

Input: String - abcdecba, k = 1

Output: Yes

String can become palindrome by removing 1 character i.e. either d or e

Input: String - abcdeca, K = 2

Output: Yes

Can become palindrome by removing 2 characters b and e (or b and d).

Input: String - acdcb, K = 1

Output: No

String cannot become palindrome by removing only one character.

2) Given a dictionary, a method to do a lookup in the dictionary and a M x N board where every cell has one character. Find all possible words that can be formed by a sequence of adjacent characters. Note that we can move to any of 8 adjacent characters, but a word should not have multiple instances of the same cell.

Example:

Input: dictionary[] = {"GEEKS", "FOR", "QUIZ", "GO"};

```
boggle[][]   = {{'G','I','Z'},
                {'U','E','K'},
                {'Q','S','E'}};
```

isWord(str): returns true if str is present in dictionary else false.

Output: Following words of the dictionary are present

GEEKS

QUIZ

3) Given k sorted arrays of possibly different sizes, find m-th smallest value in the merged array.

Examples:

Input: m = 5

arr[][] = { {1, 3}, {2, 4, 6}, {0, 9, 10, 11}} ;

Output: 4

Explanation The merged array would be {0 1 2 3 4 6 9 10 11}. The 5-th smallest element in this merged array is 4.

Input: m = 2

arr[][] = { {1, 3, 20}, {2, 4, 6}} ;

Output: 20

4) Say you have an array, A, for which the ith element is the price of a given stock on day i.

Design an algorithm to find the maximum profit.

You may complete as many transactions as you like (i.e., buy one and sell one share of the stock multiple times).

However, you may not engage in multiple transactions at the same time (ie, you must sell the stock before you buy again).

Input Format:

The first and the only argument is an array of integer, A.

Output Format:

Return an integer, representing the maximum possible profit.

Constraints:

1 <= len(A) <= 1e5

1 <= A[i] <= 1e7

Example:

Input 1:

A = [1, 2, 3]

Output 1:

2

Explanation 1:

=> Buy a stock on day 0.

=> Sell the stock on day 1. (Profit +1)

=> Buy a stock on day 1.

=> Sell the stock on day 2. (Profit +1)

Overall profit = 2

Input 2:

A = [5, 2, 10]

Output 2:

8

Explanation 2:

=> Buy a stock on day 1.

=> Sell the stock on on day 2. (Profit +8)

Overall profit = 8

5) You are given two linked lists representing two non-negative numbers. The digits are stored in reverse order and each of their nodes contain a single digit. Add the two numbers and return it as a linked list.

Input: (2 -> 4 -> 3) + (5 -> 6 -> 4)

Output: 7 -> 0 -> 8

342 + 465 = 807

Make sure there are no trailing zeros in the output list

So, 7 -> 0 -> 8 -> 0 is not a valid response even though the value is still 807.

6) Rearrange a given array so that Arr[i] becomes Arr[Arr[i]] with O(1) extra space.

Example:

Input: [1, 0]

Output: [0, 1]

Let's say N = size of the array. Then, following holds true:

1. All elements in the array are in the range [0, N-1]
2. N * N does not overflow for a signed integer

7) Given an array A of integers, find the index of values that satisfy A + B = C + D, where A, B, C & D are integers values in the array

Note:

1. Return the indices `A1 B1 C1 D1`, so that

A[A1] + A[B1] = A[C1] + A[D1]

A1 < B1, C1 < D1

A1 < C1, B1 != D1, B1 != C1

2. If there are more than one solutions, then return the tuple of values which are lexicographical smallest.

Assume we have two solutions

S1: A1 B1 C1 D1 (these are values of indices int the array)

S2: A2 B2 C2 D2

S1 is lexicographically smaller than S2 iff

A1 < A2 OR

A1 = A2 AND B1 < B2 OR

A1 = A2 AND B1 = B2 AND C1 < C2 OR

A1 = A2 AND B1 = B2 AND C1 = C2 AND D1 < D2

Example:

Input: [3, 4, 7, 1, 2, 9, 8]

Output: [0, 2, 3, 5] (O index)

If no solution is possible, return an empty list.