

CS222 Homework 4

Exercises for Algorithm Design and Analysis by Li Jiang, 2016 Autumn Semester

1. Given a non-empty integer array, find the minimum number of moves required to make all array elements equal, where a move is incrementing a selected element by 1 or decrementing a selected element by 1.

You may assume the array's length is at most 10,000.

Example:

Input: [1,2,3]

Output: 2

Explanation:

Only two moves are needed (remember each move increments or decrements one element):

$[1, 2, 3] \Rightarrow [2, 2, 3] \Rightarrow [2, 2, 2]$

Input:

int A[]: the input array.

int N: length of A.

Output:

int minMoves.

2. Given a string that consists of only uppercase English letters, you can replace any letter in the string with another letter at most k times. Find the length of a longest substring containing all repeating letters you can get after performing the above operations.

Note: Both the string's length and k will not exceed 104.

Example:

Input:

s = "AABABBA", k = 1

Output:

4

Explanation:

Replace the one 'A' in the middle with 'B' and form "AABBBBA". The substring "BBBB" has the longest repeating letters, which is 4.

Input:

string s;

int k;

Output:

return the length of the longest substring.

3. You are given an array x of n positive numbers. You start at point $(0,0)$ and moves $x[0]$ metres to the north, then $x[1]$ metres to the west, $x[2]$ metres to the south, $x[3]$ metres to the east and so on. In other words, after each move your direction changes counter-clockwise.

Write a one-pass algorithm with $O(1)$ extra space to determine, if your path crosses itself, or not.

Example 1:

Given $x = [2, 1, 1, 2]$

Return true (self crossing)

Example 2:

Given $x = [1, 2, 3, 4]$

Return false (not self crossing)

Example 3:

Given $x = [1, 1, 1, 1]$

Return true (self crossing)

Input:

int $x[]$: the input array.

int N : length of x .

Output:

return true or false.