Candies

round D

kick start 2018

keyword

Two pointers, multiset

Requirements

An array of size N. (s)

Try to find a continuous subarray satisfying

① sum of that subarray $\leq$ D → prefix sum + multiset

② Number of odd numbers $\leq$ 0 → two pointers

Solution

① Two pointer + multiset

Init: long long ans = -(1e17),

int right = 0;

int numberofOddNumbers = 0;

multiset<long long> s;

For every left in [0, N-1]:

find the index of y, satisfying requirement ②. And for every right satisfying requirements 2 put sum[right] in.

The numbers are in [left, right)

if left == right:

set right = left + 1

else:

The numbers are [left, right]

sum[right] - sum[left] + S[left] $\leq$ D

sum[right] $\leq$ D + sum[left] - S[left]

sum[right] < D + sum[left] - S[left] + 1

s.lower_bound (↓)

update ans.

Code: ipod-S.cpp