

std open Gold @ 26 Mar 2020

1. dp on id and net changes?

$\Rightarrow \text{op}(i, S, c) = \text{least waste after capturing}$

.+ i id groups and using c changes

and net of size  $S - \dots$

~~choose partition NO  
partition id groups -> size  $S - \dots$~~

if sum - size  $c = 0$ :

return  $\text{waste}_i + \text{op}(i+1, S, c)$

choose  $i$  in  $N$  id groups  $\rightarrow$

else choose  $i$  groups

else

maximize probability after partition  $\rightarrow$

then  $\text{max} = \text{op}(i+1, k, c-1)$

for all

for  $\text{size}_i \leq k \leq \text{max after } i$

choose preferred

2.  $\leftarrow$   $\text{max} \Rightarrow$  bin search? larger than ans,

so it is  $N^2 \log N$  time | hard due to constraint of groups

3. score 15:56

\* see next page both subarrays have no inversions  
+ sorted by sum

Swapping two of the same numbers does nothing

+ swap with next pair up-right then

gives 3 inversions  $n \log n$

$01 \rightarrow 10 + 1$

- inversions per 1 =

$001 \rightarrow 020 + 2$

# 0s after it.

$00011 \rightarrow 00101 + 1$

- an internal swap

$00011010$

- can only increase

$543222111$

- or decrease inversions



by 1

$00010110$

- Swapping a prefix  
com of zeros is  $O(1)$

- Swapping the boundary only adds/removes

ones before end  
zeros after begin

on left  
on right

- Completing  $\propto$  useful boundary swaps

creates/removes  $\propto$  pre-inverts.

where pre is # of ones in the org  
left side.

divide and conquer?

It will feel like a graph problem:-

but there are just too many states

g1901

## Snakes

bottom half

groups

-S, P

op(i, size, changes):

if (i == end) return group[i] ≤ size?

(W, S)

✓ waste, X INF

new = max (ref, op(i+1, size, changes))

if no change → + waste

for new size = size<sub>k</sub> for k > i  
group[i] ≤ new ≤ max after i:

new max = op(i+1, new, changes1)

16:22 N \* max a<sub>i</sub> = too large  
N/2  $\checkmark$   $\frac{N}{2}$  little slow

g1902

how to check?

never fails: Put cow i in any group that works

if no group collides: ?

if one group collides: Put in that group

if two groups collide: impossible!

Conditioned

g1902

$n^2$  check

: (expands)  $\dots \times \dots \times \dots$

checks (~~mindest~~) :  $\dots \times \dots$

init d's  
for. (i, N)

((expands)  $\dots \times$  for (j, i))  $\dots = \dots$

if (check (i, j)) ~~mindest~~ :  
: & (diff (i, j))  
: merge (i, j)  
-- groups

higher mindest  $\rightarrow$  condition more satisfied

17:54: order  $\rightarrow$  less groups  
of d's matters!

1. good: i is won for:  $\dots \times \dots$

elsewise group

2. : 2nd max group on top

with i: 2nd max group on top

group

3. elimination:  $\dots \times \dots \rightarrow$  no group