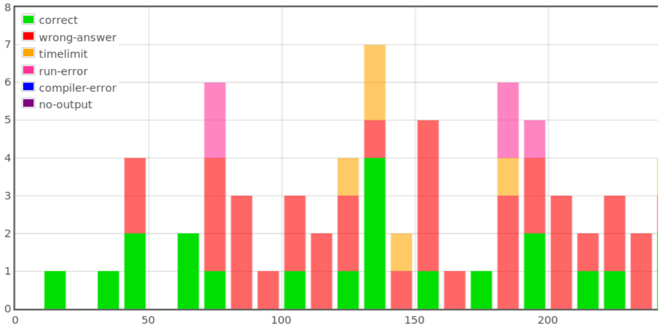
K – Dishonest Driver

Solved by 18 teams before freeze.
First solved after 17 min by **Team RacIETH**.



K – Dishonest Driver

Problem

Given a string, compute the length of its shortest compressed form.
How to build a compressed form:

- one character c (size: $|c| = 1$),
- concatenation $w_1 w_2$ (size: $|w_1 w_2| = |w_1| + |w_2|$),
- repetition $(w)^n$ (size: $|(w)^n| = |w|$).

K – Dishonest Driver

Solution in time $\mathcal{O}(N^3)$

Dynamic programming on:

$$F(i,j) = \text{size of compressed form of substring } u_{ij} = u_i \dots u_{j-1}$$

If $j = i + 1$, then $F(i,j) = 1$. Otherwise:

- Try splitting $u_{ij} = u_{ik} u_{kj}$ for any position $k \in [i+1, j-1]$;
- Try factorizing u_{ij} into $u_{ij} = u_{ik}^n$:
  - What are the factorizations of u_{ij}?
  - Trick: search second occurence of u_{ij} in $u_{ij} u_{ij}$
  - $\mathcal{O}(N)$ with KMP (e.g., use C++ stdlib `find` function)

Note: we also have a $\mathcal{O}(N^2 \log N)$ algorithm