Commentaries on Problems

ICPC 2025 ASIA YOKOHAMA REGIONAL JUDGE TEAM

(CHIEF: MITSURU KUSUMOTO)

Judge Team

Takashi Chikayama (U Tokyo)

Shuichi Hirahara (NII)

Kazuhiro Inaba (Google)

Kiyoshi Ishihata (Meiji U)

Yuya Kadono (Kyoto U)

Masatoshi Kitagawa (Kyushu U)

Koichi Kubota (Chuo U)

Soh Kumabe (Cyber Agent)

Mitsuru Kusumoto (Monoxer)

Naoki Marumo (U Tokyo)

Kentaro Matsushita (Goldman Sachs Japan)

Kiminori Matsuzaki (Kochi U of Tech)

Tomohiro Oka (Google)

Ryotaro Sato (Preferred Networks)

Etsuya Shibayama (ROIS)

Shinya Shiroshita (Preferred Networks)

Tatsuya Sumiya (MCD3)

Tomoharu Ugawa (U Tokyo)

Fumihiko Yamaguchi (Toyota Tech Institute)

Easy ⟷ Judge's estimation ⟷ Hard

	D	E	H	J	I	C	A	L	B	G	F	K
#Corrects	63	61	57	30	24	16	25	2	1	10	0	0

Harder than our guess?

↑😲

First Acceptance

A	68min	O1_Practice
B	268min	Rinshan Solution
C	30min	Rinshan Solution
D	6min	The Revenge of take000
E	22min	Strong Zero
F	???min	???

G	104min	The Revenge of shinchan
H	31min	Strong Zero
I	56min	The Revenge of shinchan
J	60min	Rinshan Solution
K	???min	???
L	179min	Strong Zero

Congratulations!!!

Full problem analysis is available

https://icpc.jp/2025/regional

We will not cover everything in this commentary.
Please see the problem analysis for detail
or directly contact with judges.

D: Decompose and Concatenate

Proposer: Kazuhiro Inaba
Author: Fumihiko Yamaguchi
Presenter: Fumihiko Yamaguchi

Problem and Solution

This is a problem where you decompose a given positive integer into a sum of two positive integers, concatenate them as strings, and find the maximum resulting number.

You can solve this by considering cases based on whether the top two digits of the given number are 10 or not.

All teams solved this problem. Congratulations!

E: Cutting Tofu

Proposer: Naoki Marumo
Author: Masatoshi Kitagawa
Presenter: Masatoshi Kitagawa

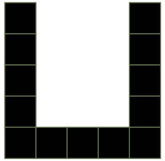
Problem and Solution

Maximize x s.t. $\lfloor a/x \rfloor \cdot \lfloor b/x \rfloor \cdot \lfloor c/x \rfloor \leq k$.

Output x as a reduced fraction.

The maximum value is of the form $a/i, b/i, \text{or } c/i$ for some positive integer i.

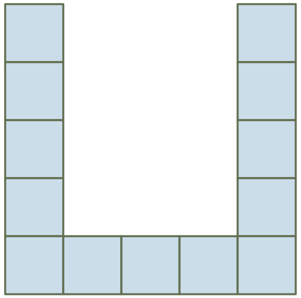
Binary search on i!

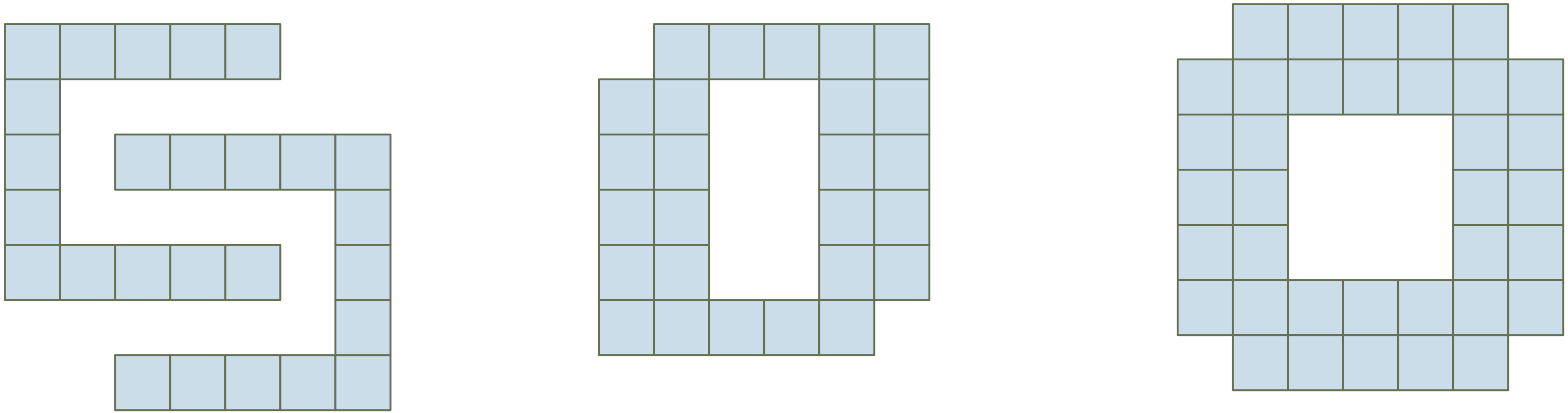
H: -Shaped Panels

Proposer: Mitsuru Kusumoto
Author: Tomoharu Ugawa
Presenter: Tomoharu Ugawa

Problem

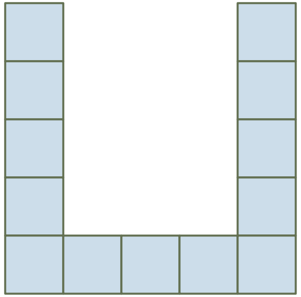
Can we arrange U-shaped panels as required?

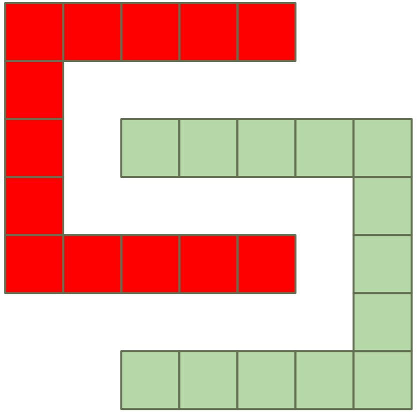


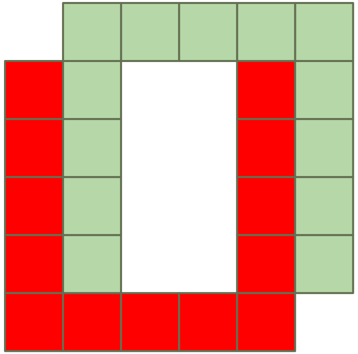


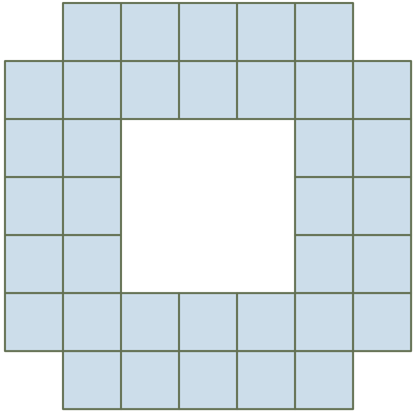
Problem

Can we arrange U-shaped panels as required?

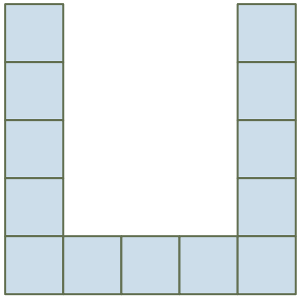


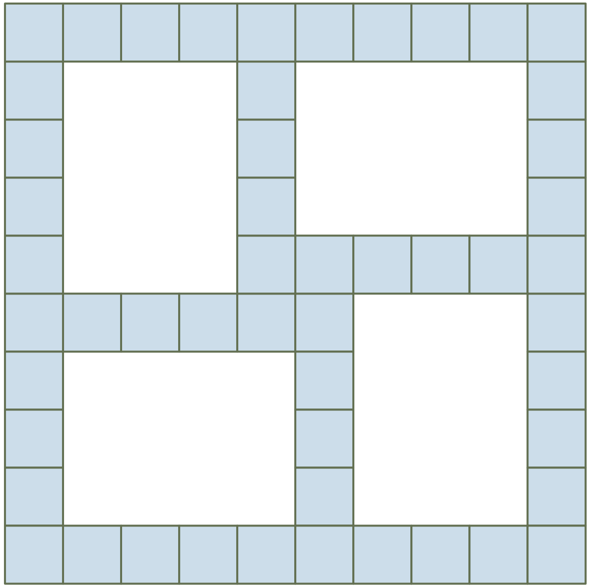




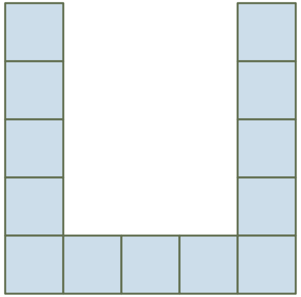


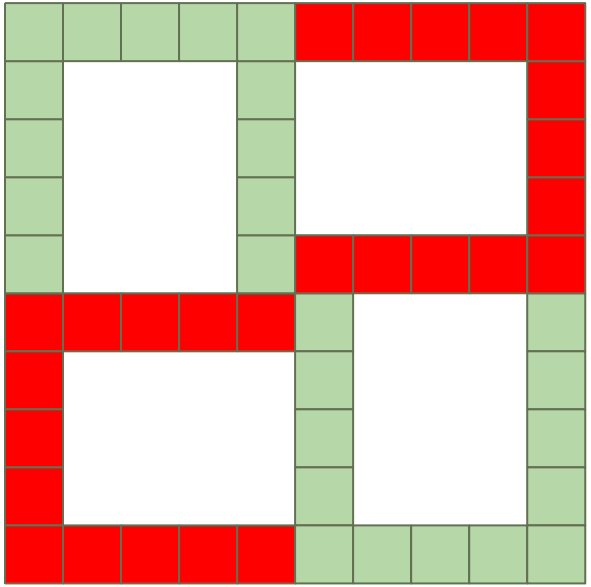
How About This?

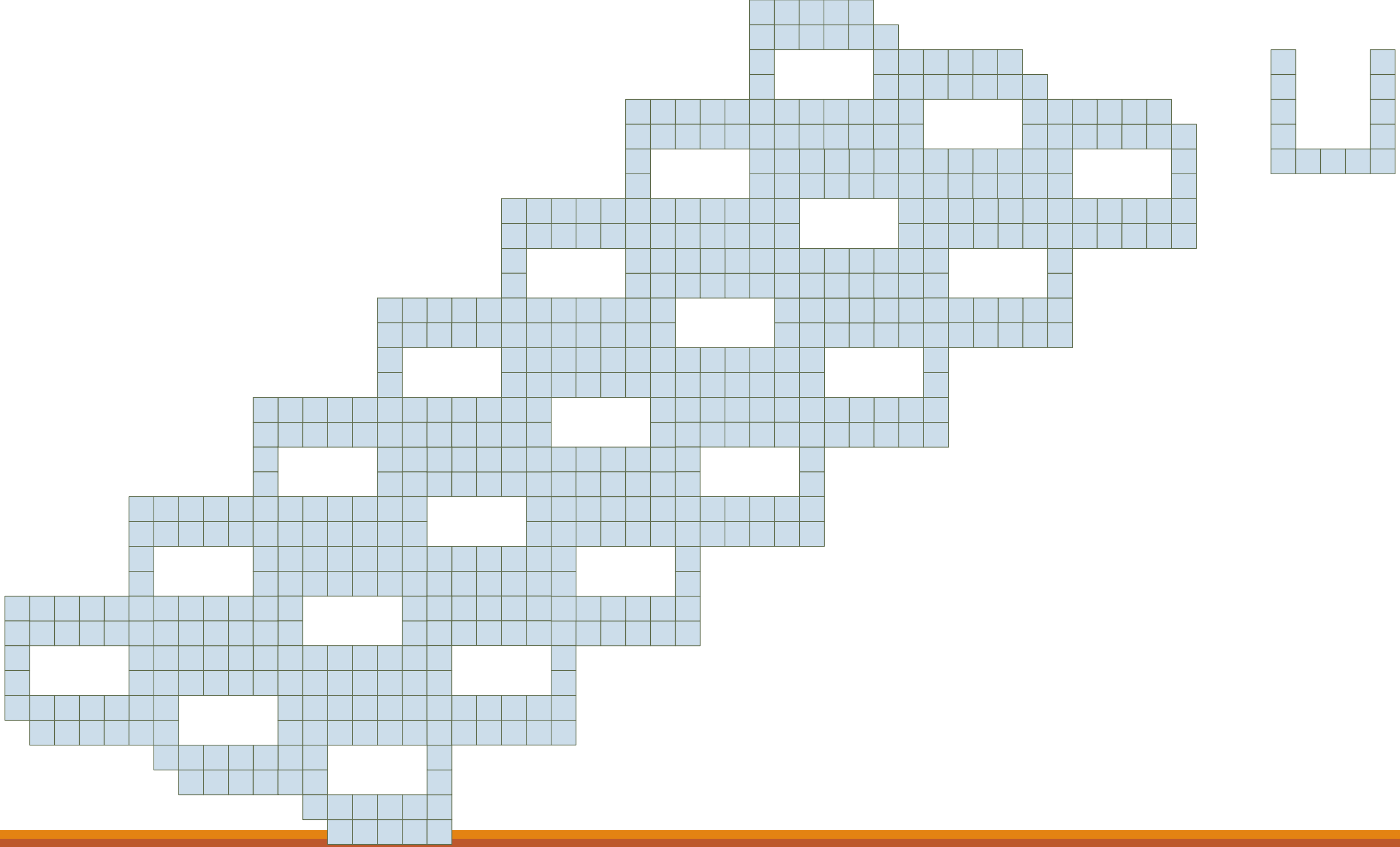


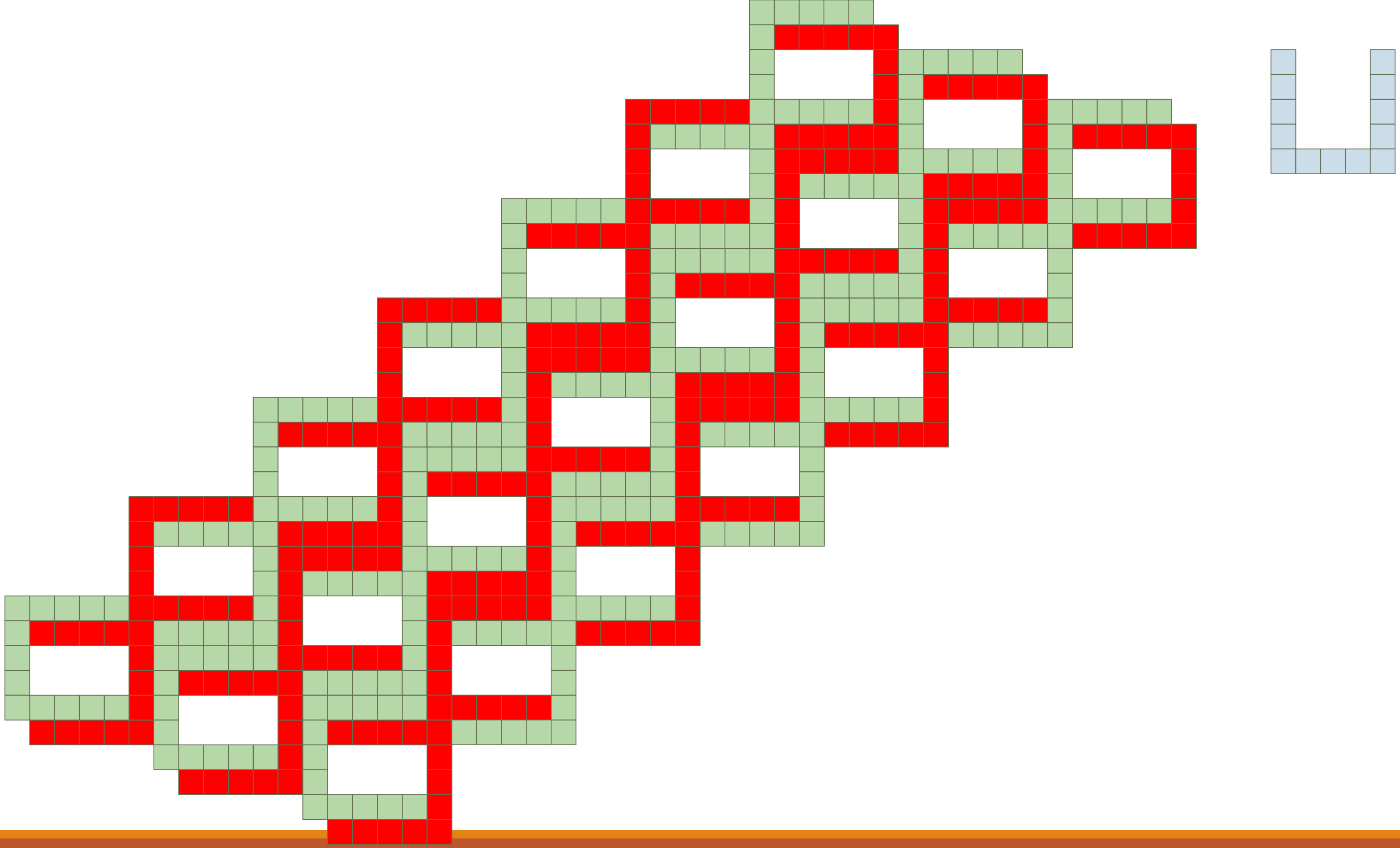


How About This?



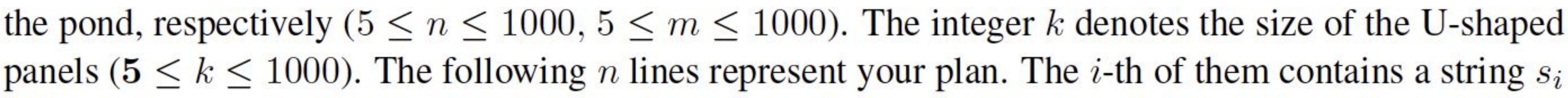






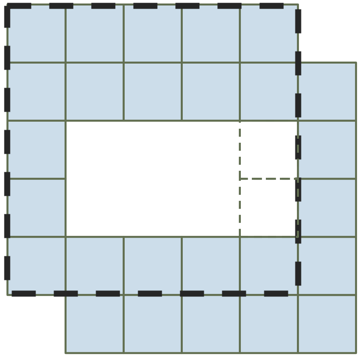
Key Observation

Lower bound of k is unnaturally emphasized in the problem text



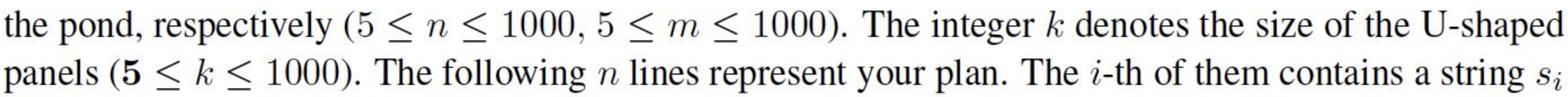
the pond, respectively ($5 \leq n \leq 1000$, $5 \leq m \leq 1000$). The integer k denotes the size of the U-shaped panels ($\mathbf{5} \leq k \leq 1000$). The following n lines represent your plan. The i-th of them contains a string s_i

The orientation of U-shaped panel can be determined uniquely



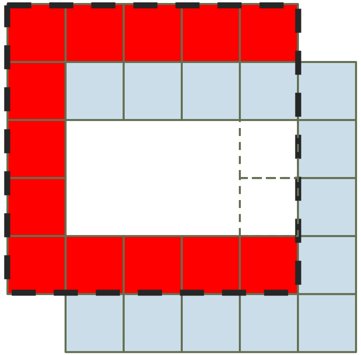
Key Observation

Lower bound of k is unnaturally emphasized in the problem text



the pond, respectively ($5 \leq n \leq 1000$, $5 \leq m \leq 1000$). The integer k denotes the size of the U-shaped panels ($\mathbf{5} \leq k \leq 1000$). The following n lines represent your plan. The i-th of them contains a string s_i

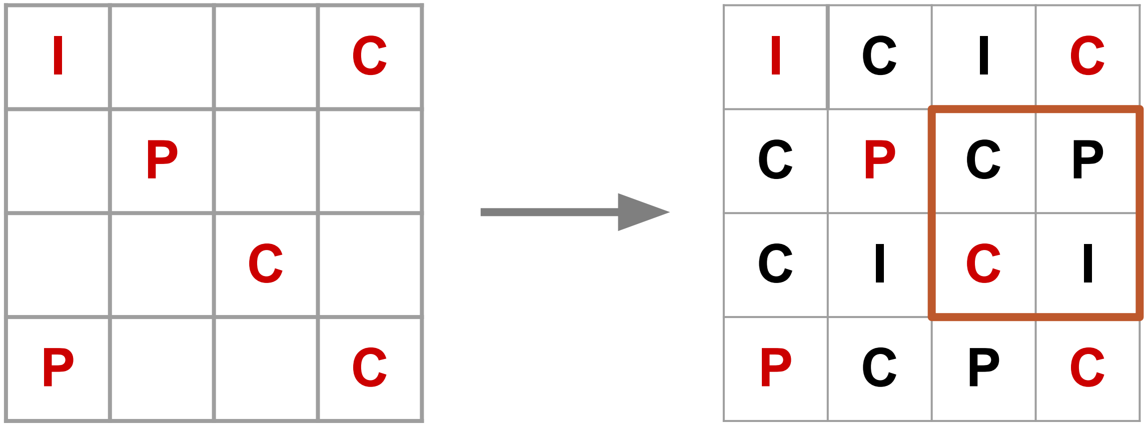
The orientation of U-shaped panel can be determined uniquely



J: ICPC Board

Proposer: Naoki Marumo
Author: Naoki Marumo
Presenter: Naoki Marumo

Problem



I			C
	P		
		C	
P			C

I	C	I	C
C	P	C	P
C	I	C	I
P	C	P	C

Every 2 x 2 region should have “ICPC”

Fill the blank cells, or Report it’s impossible

Split into cases **carefully** 🧐

All possible patterns

C	I	C	I	…
C	P	C	P	…
I	C	I	C	…
C	P	C	P	…
	⋮			

or

I	P	I	P	…
C	C	C	C	…
P	I	P	I	…
C	C	C	C	…
	⋮			

or their transposes

Coding effort may vary greatly by approach 🛠️

I: Game of Names

Proposer: Tatsuya Sumiya
Author: Soh Kumabe
Presenter: Tatsuya Sumiya

Problem

- Alice and Bob play a game on a board consisting of cells arranged in a row.
- Some cells initially contain one of the players' names.
- Starting with Alice, the two players take alternately write their own name in a blank cell. **Cells adjacent to a cell containing their own name are not allowed.**
- The player who cannot make a move loses.
- Determine the winner for the given initial board state.

	Bob				Alice		

Key Idea

Suppose that the winner writes their own name as possible even after the game ends.

The examples of final states are:

Alice	Bob		Alice	Bob	Alice	Bob	Alice

Alice	Bob	Alice	Bob		Alice		Bob

“Alice” and “Bob” appear alternately!

Solution

- The final difference in the number of “Alice” and “Bob” depends only on which player takes each of the end cells.
  $\longrightarrow$ Both players should aim for the end cells first.
- It is sufficient to consider the optimal actions of two players until both end cells are taken.

C: Seagull Population

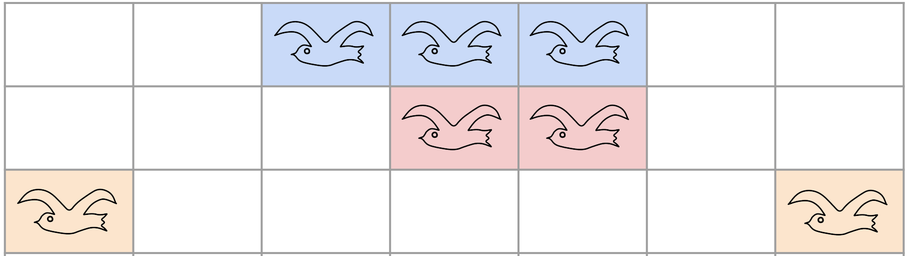
Proposer: Shinya Shiroshita
Author: Shinya Shiroshita
Presenter: Shinya Shiroshita

Problem

You are given the daily counts of seagulls staying in an island.

- Each seagull visits and leaves the island exactly once per year. Its stay period may across years.

Calculate the minimum number of seagulls (and an example) satisfying the case.



#(seagulls)	1	0	1	2	2	0	1

Sample visiting schedules

Lower bounds

There are two lower bounds.

- $D_{\text{max}} = \max(b_1, \ldots, b_n)$: The maximum number of seagulls observed in one day.
- $S_{\text{inc}} = \sum_{i=1,\ldots,n} \max(b_i - b_{i-1}, 0)$ $(b_0 = b_n \text{ for simplicity})$: The total number of increasing seagull numbers.
  ($\because$ There must be $b_i - b_{i-1}$ appearing seagulls on day i .)

Now you obtain a lowerbound $M_{\text{opt}} = \max(D_{\text{max}}, S_{\text{inc}})$.

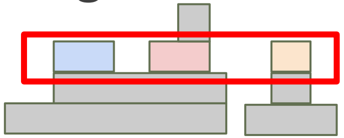
$\longrightarrow$ In fact, this value is ***the optimal!***

Construction

If there is no seagull on some day, greedily assign latter departure (First-In-Last-Out) from the zero day can get an optimal solution.

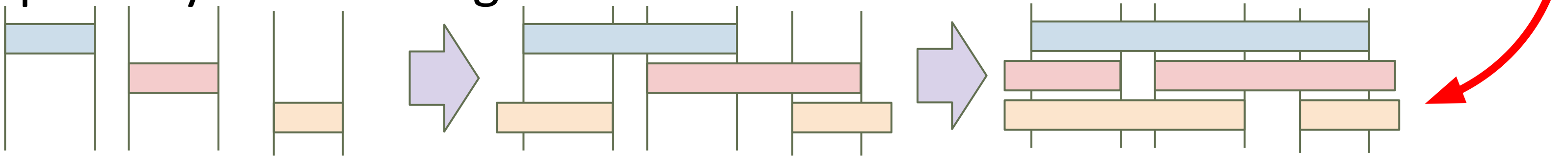
For the case daily minimum > 0, construct above-minimum schedules and reduce additional seagulls by the following idea.

Key idea: Rotate departure timing among seagulls whose schedules don't overlap.



FILO construction

Grouping by heights and rotating departure timings inside groups can optimally reduce seagulls!

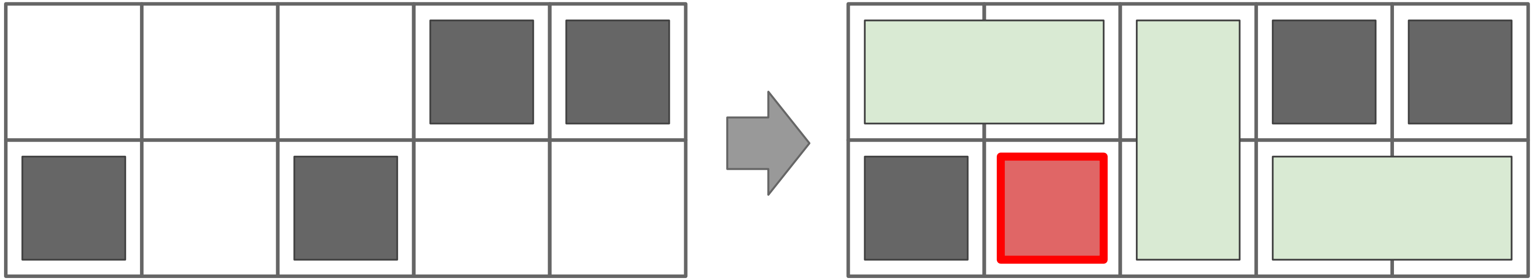


A: Tatami Renovation

Proposer: Ryotaro Sato
Author: Ryotaro Sato
Presenter: Ryotaro Sato

Problem

- Grid of 2 rows
- Some ($\leqq 10^5$) blocks are on distinct cells
- Move **minimum** number of blocks to **adjacent** cells to cover all the empty cells with 2×1 dominos

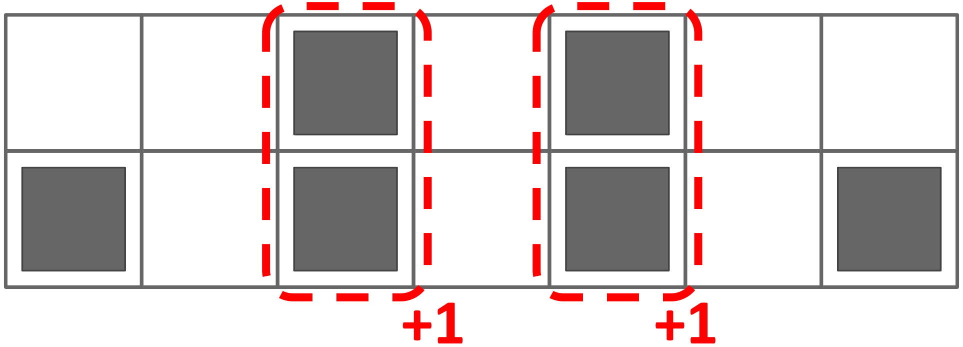


Solution

Calculate the lower bound of the answer.

First, split given grid at every point where the number of blocks to the left is even.

Obviously, if a column in a segment contains two tiles, at least one of these must be moved.

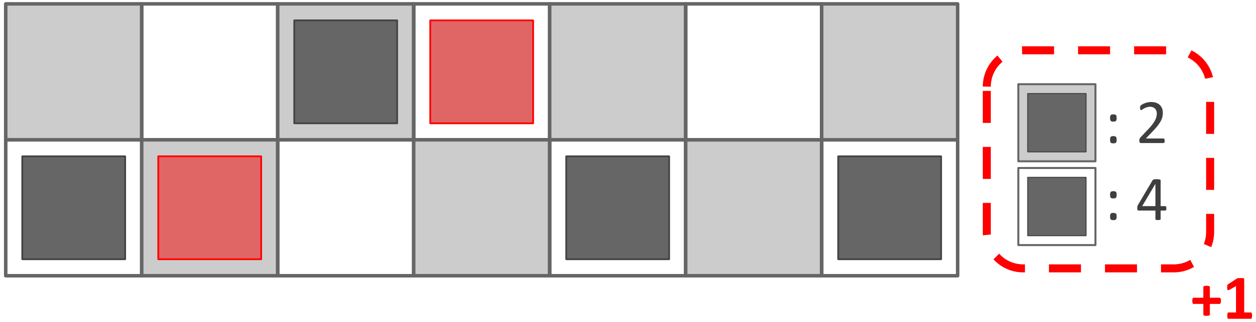


Solution

Paint the grid like chessboard.

If # of tiles to be on “black” cells and “white” cells in a segment differ, one more move is required (proof omitted).

This lower bound is always achievable (proof omitted).

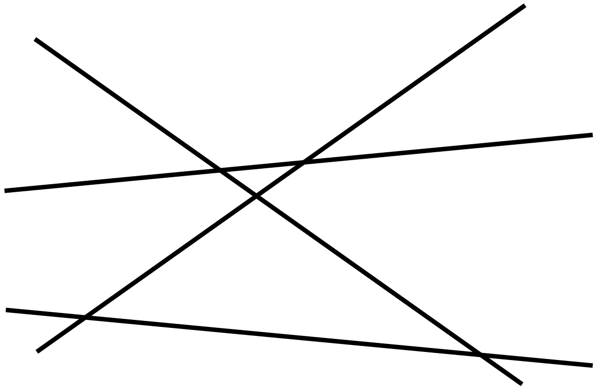


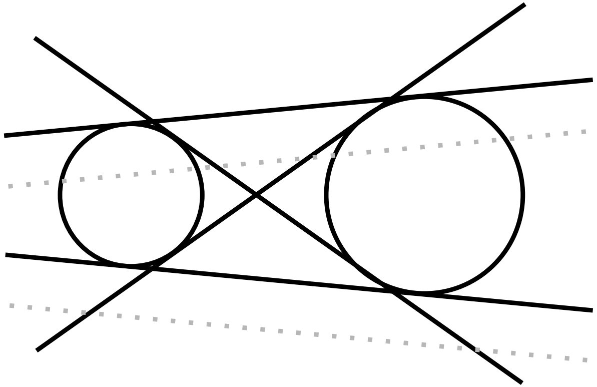
L: Common Tangent Lines

Proposer: Mitsuru Kusumoto
Author: Mitsuru Kusumoto
Presenter: Tomohiro Oka

Problem

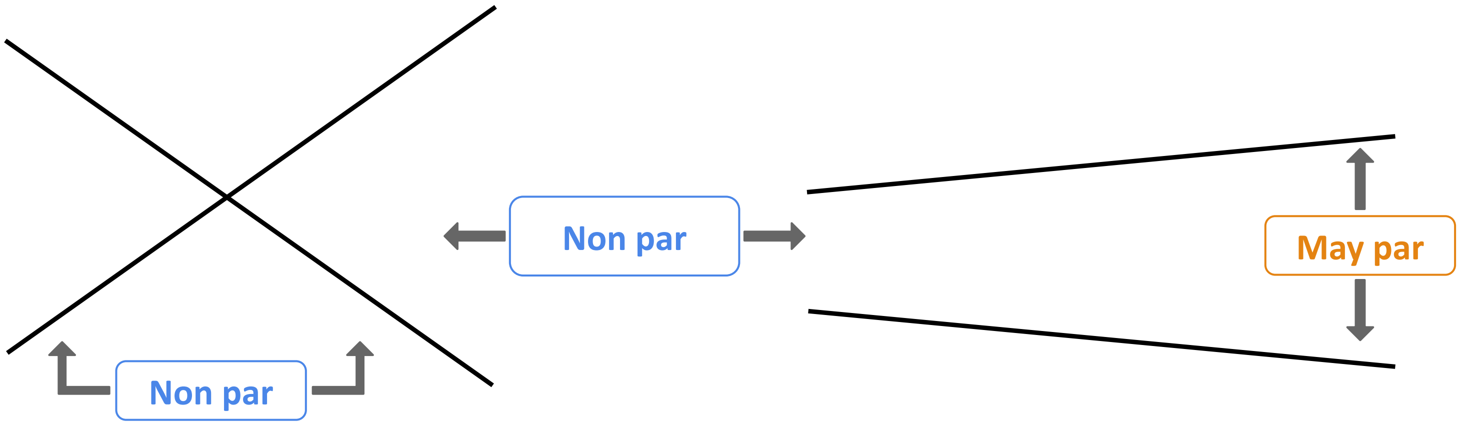
- Given 4 lines
- We can move the lines (translation, not rotation)
- Are there 2 circles with the 4 common tangent lines?
  - if yes, find minimum total distance of translation





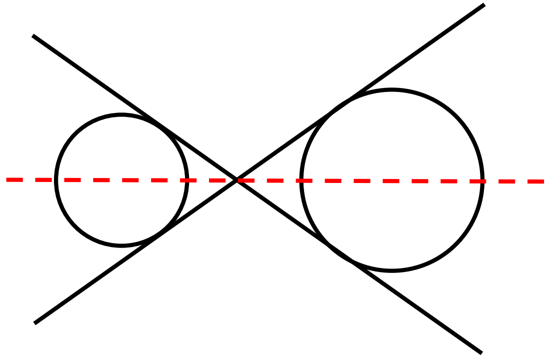
Partition 2 pairs

- Try all combinations of line pairs
  - Internal and external tangent lines
- Check non-parallel conditions
  - External lines may parallel
  - The other pairs must not parallel



Symmetric axes

- The centers of the circles must lie on the symmetric axis (angle bisector line)

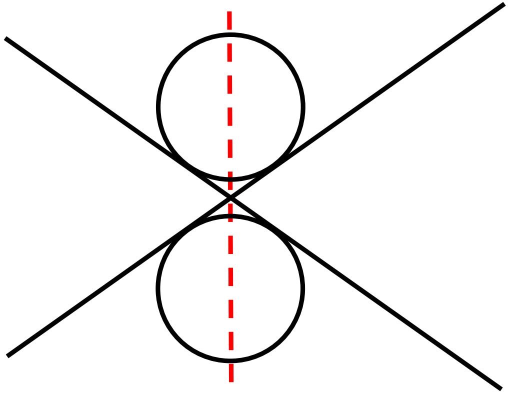


$$x \cos\left(\frac{\pi\alpha}{180}\right) + y \sin\left(\frac{\pi\alpha}{180}\right) = \frac{d_i + d_j}{2\cos\left(\frac{\pi\beta}{180}\right)}$$

$$\alpha = \frac{a_i + a_j}{2} \qquad \beta = \frac{a_i - a_j}{2}$$

Symmetric axes

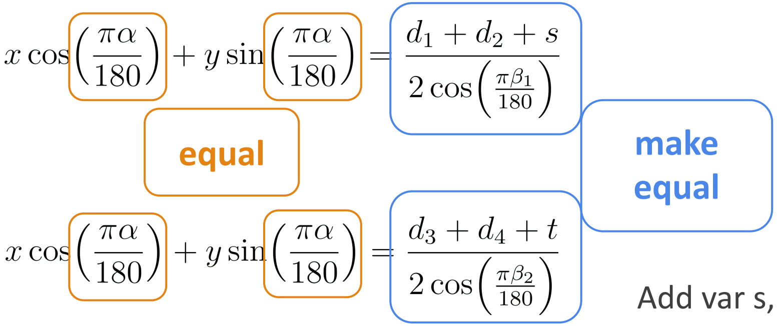
- Another angle bisector line, if two lines are crossing



$$x\cos\left(\frac{\pi(\alpha+90)}{180}\right)+y\sin\left(\frac{\pi(\alpha+90)}{180}\right)=\frac{d_i-d_j}{2\sin\left(\frac{\pi\beta}{180}\right)}$$

Match 2 axes

- Select one axis from each group
- We don’t need to translate multiple input lines



axis-1:

$$x\cos\boxed{\left(\frac{\pi\alpha}{180}\right)}+y\sin\boxed{\left(\frac{\pi\alpha}{180}\right)}=\boxed{\frac{d_1+d_2+s}{2\cos\left(\frac{\pi\beta_1}{180}\right)}}$$

equal

make equal

axis-2:

$$x\cos\boxed{\left(\frac{\pi\alpha}{180}\right)}+y\sin\boxed{\left(\frac{\pi\alpha}{180}\right)}=\boxed{\frac{d_3+d_4+t}{2\cos\left(\frac{\pi\beta_2}{180}\right)}}$$

Add var s, when d1 or d2 are moving.
Add var t, when d3 or d4 are moving.
The answer is minimum of |s| or |t|.

B: Minimizing Wildlife Damage

Proposer: Mitsuru Kusumoto
Author: Kentaro Matsushita
Presenter: Kentaro Matsushita

Problem Overview

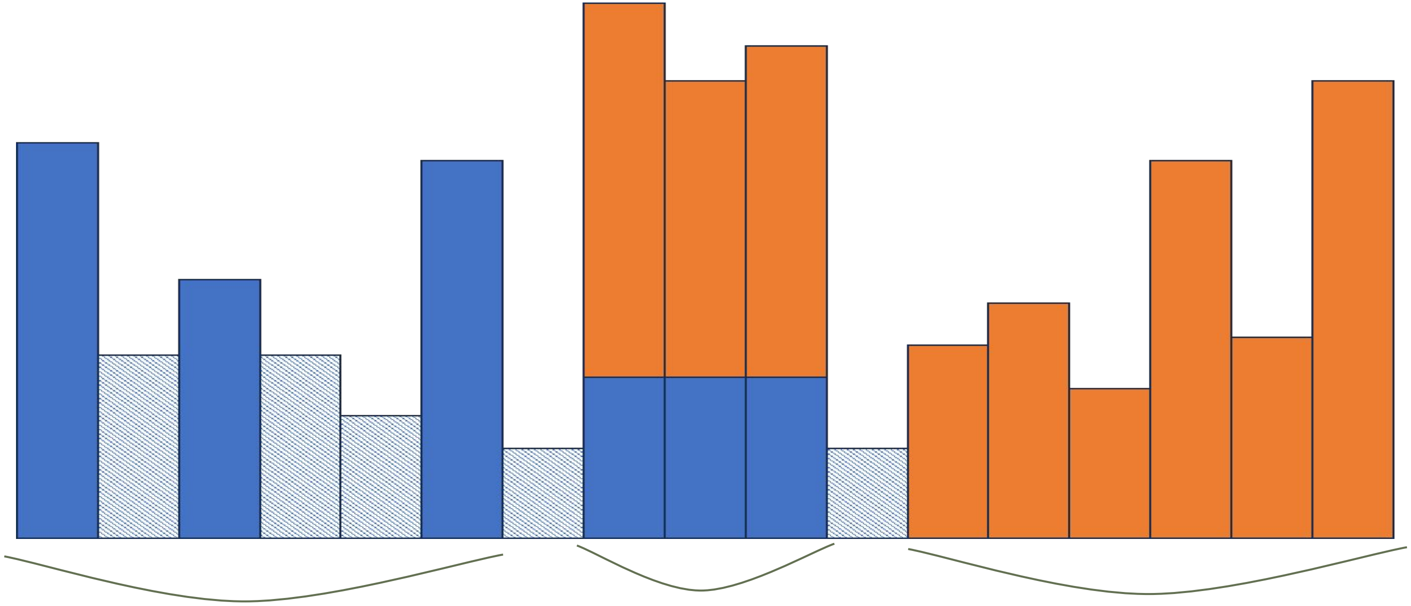
sequence A = (a_1, a_2, …, a_N) is given.

For query, integer d is given.

After changing some a_i's to 0, decrease 1 from left most consecutive block which satisfies a_i > 0.

Find optimal way so that sum of remaining a_i to be maximized.

Structure of Optimal Solution



part 1

part 2

part 3

Structure of Optimal Solution

- part1 : easily can compute by simple DP
  - never set 3 consecutive components into 0

- part2 : set one more point to 0
  - remaining number can be compute by Convex Hull Trick

- Point: can maximize part1 to consume d
  - First, compute how long is used to part1.
  - Second, scanning from back to front with maintaining CHT.

G: Charity Raffle

Proposer: Tatsuya Sumiya
Author: Tatsuya Sumiya
Presenter: Tatsuya Sumiya

Problem

We have an integer sequence $A=(a_1,\ldots,a_n)$, where all elements are initialized as $a_i=0$.

Perform the following operation k times:

Generate two random integers i and j ($1 \leq i < j \leq n$), and

- $a_i \leq a_j \rightarrow$ Increment a_i
- $a_i > a_j \rightarrow$ Increment a_j

Find the number of possible sequences as A s.t. $\max(A) \leq m$.

Conditions for Sequence A

Clearly, sum(A) = k is required.

In addition, single large element cannot exist.
e.g.) A = (1, 0, 4, 2) is impossible.

More precisely, it’s impossible if …

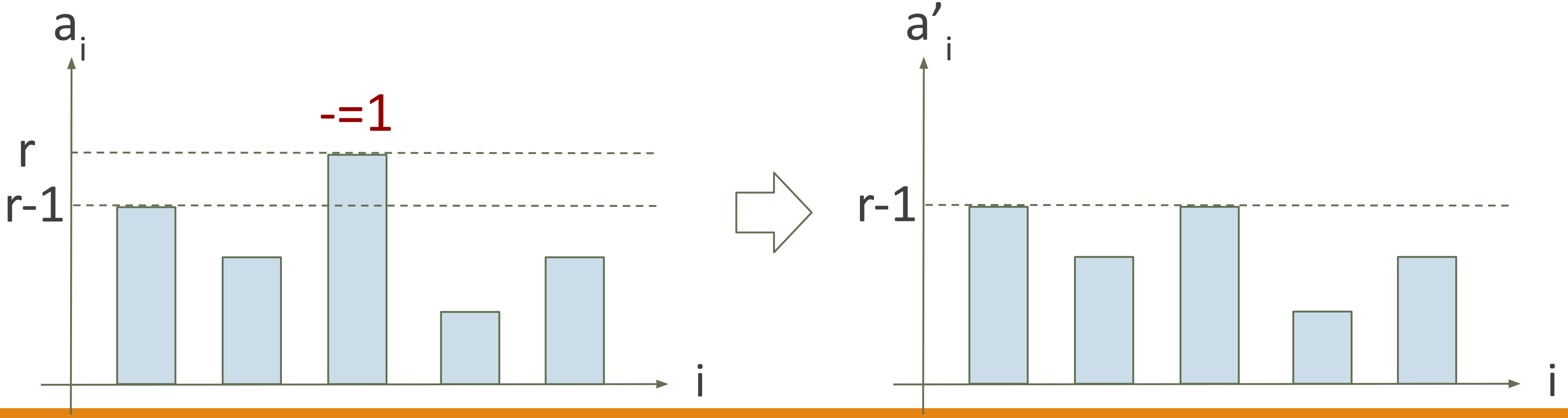
- Sequence A has only one maximum element $a_q = r$
- $a_i < r - 1$ holds for all $i > q$

Otherwise, we can prove it’s possible!

How to Count Impossible A

Consider a sequence A’ obtained by decrementing the maximum element of A ($a_q = r \longrightarrow a'_q = r - 1$). Then,

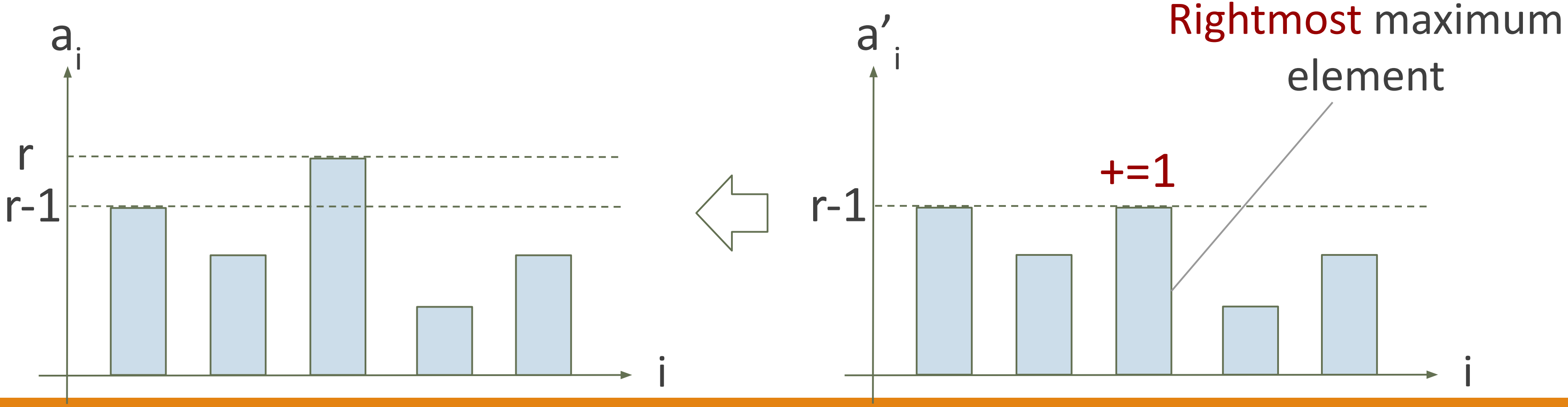
- sum(A’) = k - 1
- max(A’) = r - 1



How to Count Impossible A

Conversely, for any sequence A’, we can obtain A with the original condition by incrementing the rightmost maximum element.

This relationship is one-to-one.
Counting appropriate A’ will solve the problem!



Another Solution

We have another solution that directly applies the principle of inclusion and exclusion.

Please see the full problem analysis for details!

F: Astral Geometry

Proposer: Naoki Marumo
Author: Mitsuru Kusumoto
Presenter: Mitsuru Kusumoto

F is...

3D geometry?

Interactive?

😨

✨Linear algebra✨

$r_i = [x_i, y_i, z_i]$: Point

🧐 Distance? $|r_i - r_j|^2$

↓

Inner product ❗ ✨$r_i \cdot r_j$✨

$$\underset{n\times 3}{\begin{bmatrix} r_1 \\ r_2 \\ \vdots \\ r_n \end{bmatrix}} \underset{3\times n}{[r_1\ r_2 \dots r_n]} = \begin{bmatrix} r_1^T r_1 & r_1^T r_2 & \dots & r_1^T r_n \\ r_2^T r_1 & r_2^T r_2 & \dots & r_2^T r_n \\ \vdots & & & \\ r_n^T r_1 & r_n^T r_2 & \dots & r_n^T r_n \end{bmatrix}$$

Measurement = determine one element

Objective = determine all of these

$$\underset{n\times 3}{\begin{bmatrix} r_1 \\ r_2 \\ \vdots \\ r_n \end{bmatrix}} \underset{3\times n}{[r_1\ r_2 \dots r_n]} = \underset{\text{rank} \leq 3}{\begin{bmatrix} r_1^T r_1 & r_1^T r_2 & \dots & r_1^T r_n \\ r_2^T r_1 & r_2^T r_2 & \dots & r_2^T r_n \\ \vdots & & & \\ r_n^T r_1 & r_n^T r_2 & \dots & r_n^T r_n \end{bmatrix}}$$

Assume r_1, r_2, r_3 are linearly independent.
Then, the 4x4 submatrix

$$\begin{bmatrix} r_1 \\ r_2 \\ r_3 \\ r_x \end{bmatrix} [r_1\ r_2\ r_3\ r_y] = \begin{bmatrix} r_1^T r_1 & r_1^T r_2 & r_1^T r_3 & r_1^T r_y \\ r_2^T r_1 & r_2^T r_2 & r_2^T r_3 & r_1^T r_y \\ r_3^T r_1 & r_3^T r_2 & r_3^T r_3 & r_3^T r_y \\ r_x^T r_1 & r_x^T r_2 & r_x^T r_3 & r_x^T r_y \end{bmatrix}$$

is **linearly dependent** for any x≠y. This observation leads to determine $r_x \cdot r_y$ for all x≠y. 3n measurements are sufficient.

K: Membership Structure of a Secret Society

Proposer: Kazuhiro Inaba
Author: Kazuhiro Inaba
Presenter: Kazuhiro Inaba

Problem: Can we **satisfy** all the given **logical formulas?**

$x1 \in R(x2)$

$x1 \notin R(x2)$

$R(x1) = R(x2) \cap R(x3)$

(and global conditions

forall $y1, y2$: $R(y1)=R(y2) \Leftrightarrow y1=y2$

no loop by “$\_ \in R(\_)$”)

Satisfying logical formulas…

SAT ?

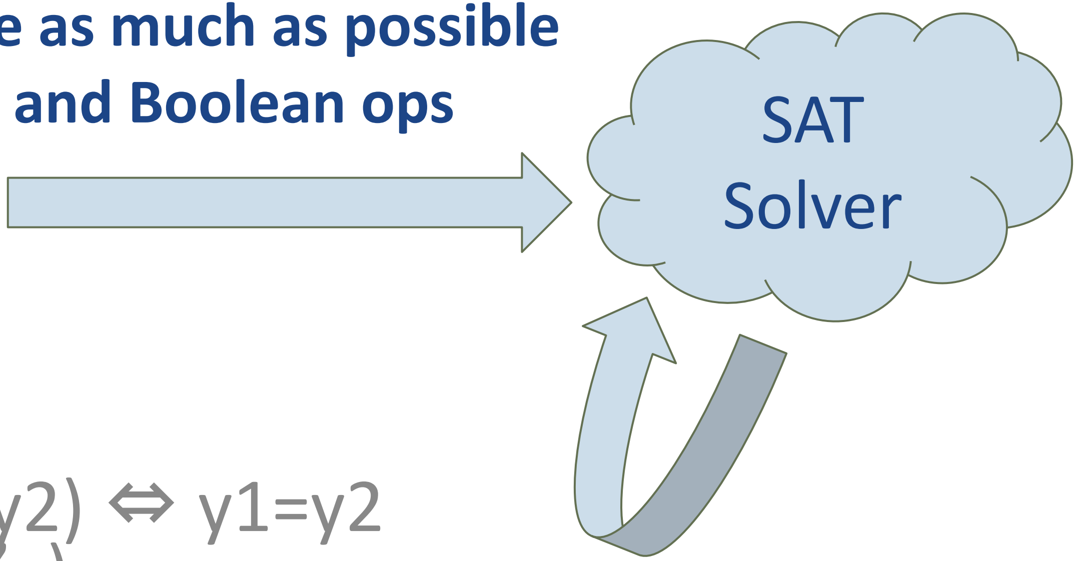
Non-Solution: “SMT Solver”

$x1 \in R(x2)$
$x1 \notin R(x2)$
$R(x1) = R(x2) \cap R(x3)$

(and global conditions

forall y1, y2: $R(y1)=R(y2) \Leftrightarrow y1=y2$
no loop by “$\_ \in R(\_)$”)

Encode as much as possible to bits and Boolean ops



SAT Solver

Guide SAT solver search with non-encodable conditions

Non-Solution: “SMT Solver”

It *may* work with a state-of-the-art SAT solver,

Or, it may be the only possible option if the inputs are very *general* logical formulas.

⇒ In this problem,
the formulas are in **very specific forms**.

Satisfying logical formulas…

Model Theory !

Model Theory (math)

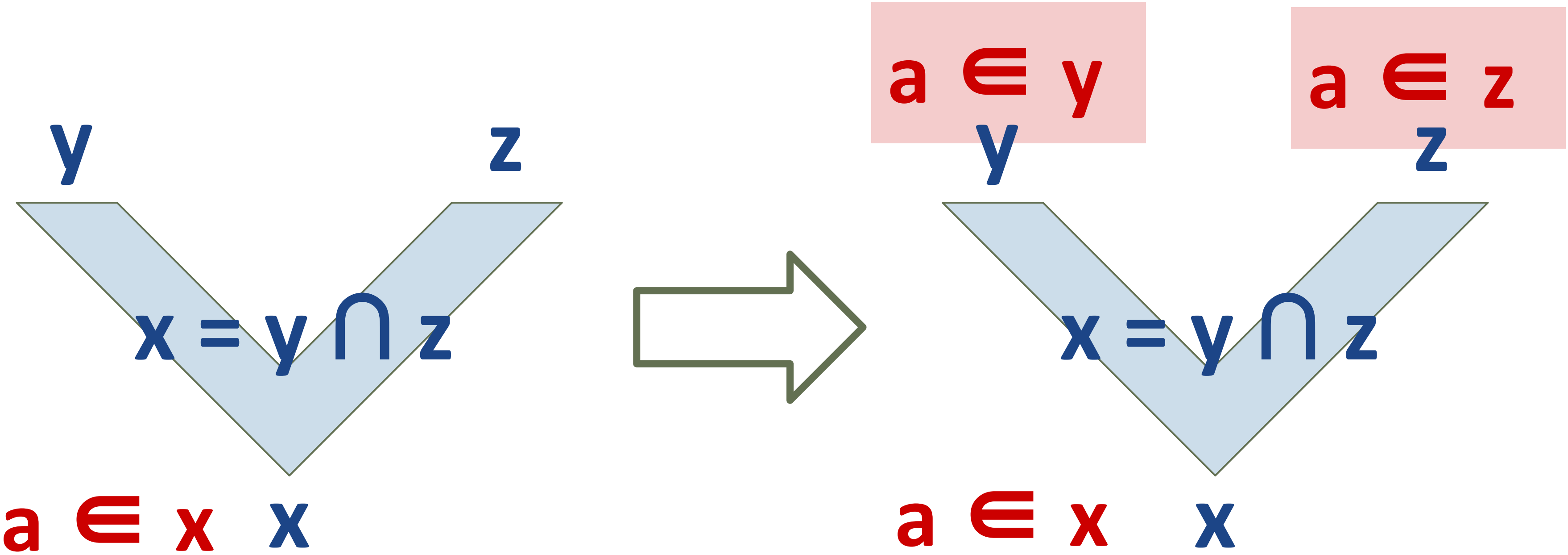
Studying relationships between logical formulas and their **models** (= structures satisfying the formulas)

Henkin’s method (wikipedia):

> *Very* roughly, **extend the input by more and more formulas** as long as they are consistent. The outcome **naturally describes a model**.

Extend by consistent formulas

Logical consequences are safe to add.



y z

$x = y \cap z$

$a \in x$ x

$a \in y$ $a \in z$

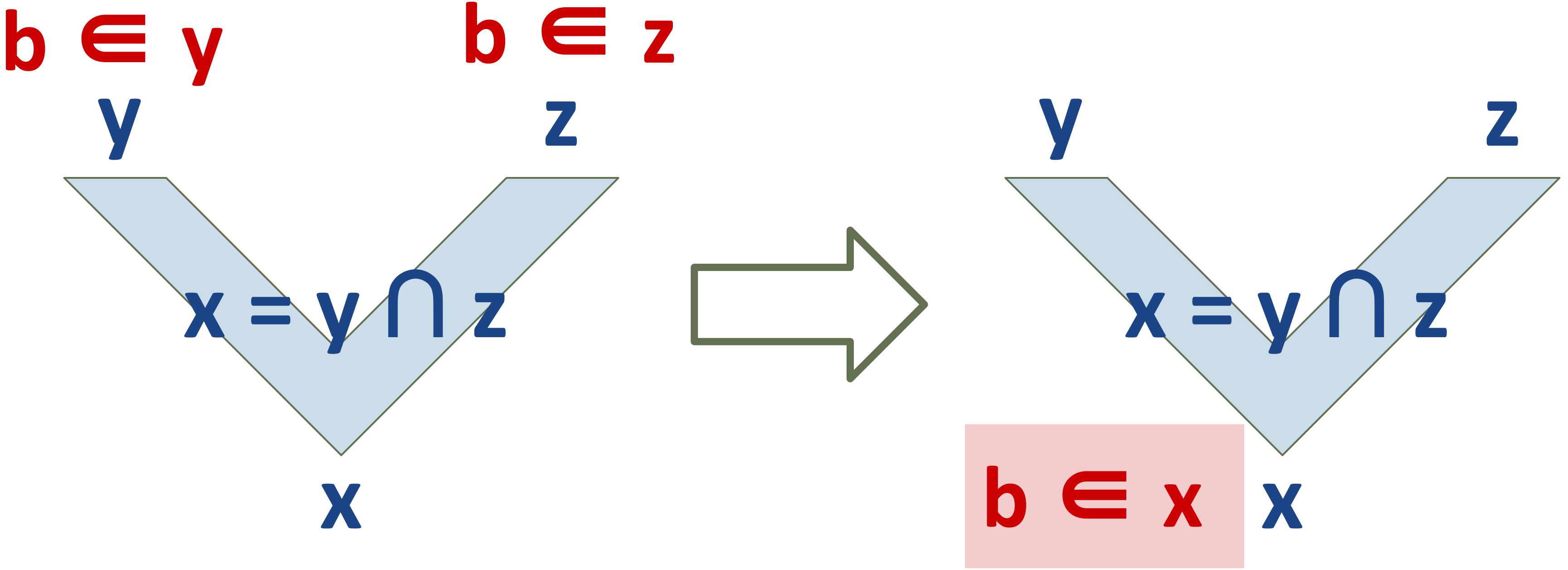
y z

$x = y \cap z$

$a \in x$ x

Extend by consistent formulas

Logical consequences are safe to add.



Extend by consistent formulas

Logical consequences are safe to add.

forall a. ($a \in x \Leftrightarrow a \in y$) ⇒ **$x = y$**

$x=y$ **$x \in z$** ⇒ **$y \in z$**

For this problem, just **adding those easy logical consequences is enough** to reconstruct the model.

Solution

(Read the problem analysis doc for detail!)

1. Generate all logical consequences (**BFS**), carefully handling “=” relations (**UnionFind**?)
2. Check if there’s contradicting $\in$ and $\notin$.
3. Check if there’s $\in$ loop (**BFS**).

EOF