

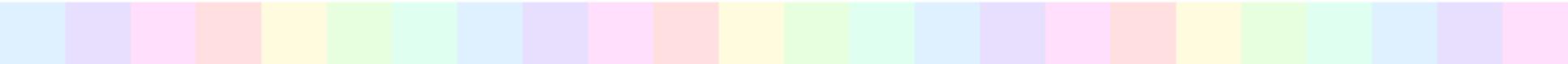
# F: Polygon Guards

JAG 春コンテスト 2014

原案：保坂

解答：保坂・田中

解説：保坂

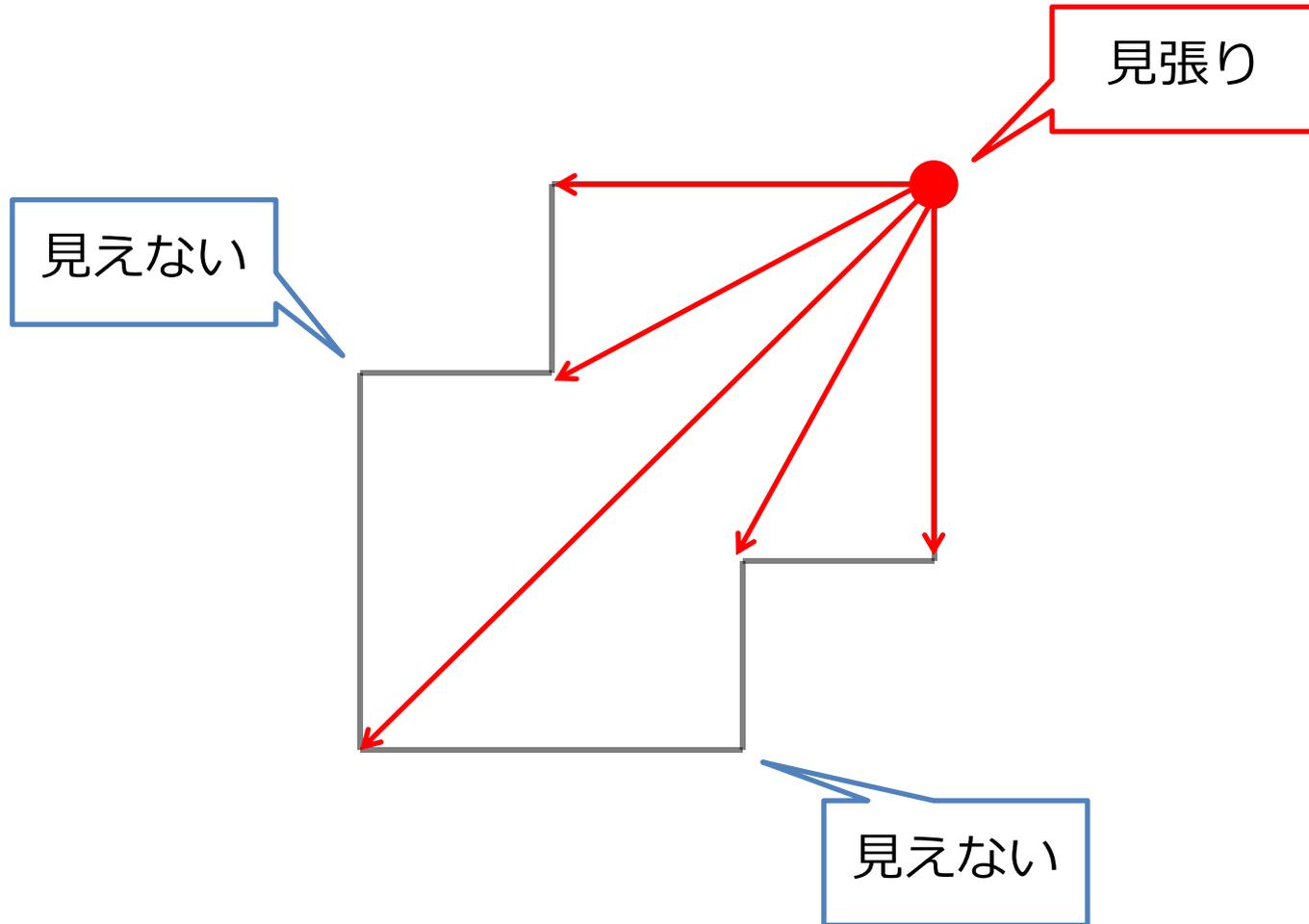


# 問題概要

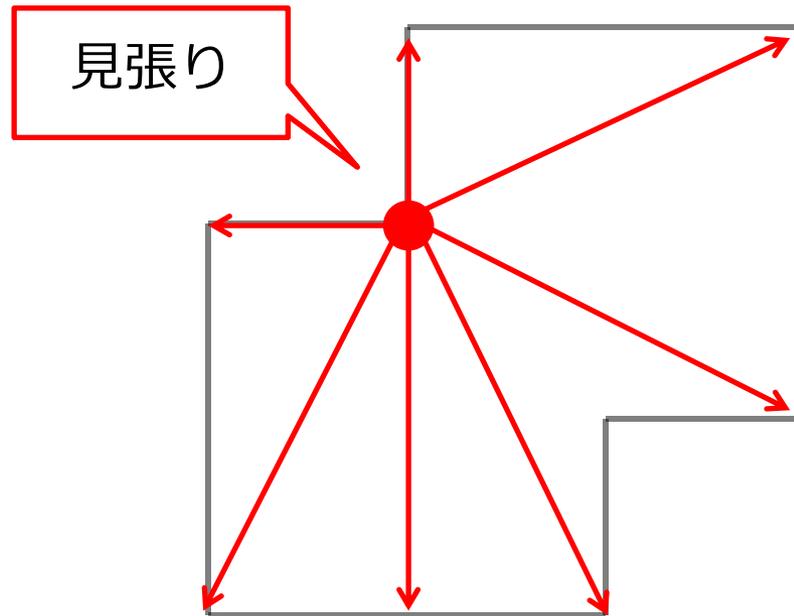
- $N$  角形 ( $4 \leq N < 40$ ) が与えられる
  - 凸とは限らない
  - 各辺は座標軸に平行
- 頂点のうちいくつかに見張りを置いて、すべての頂点を監視したい
  - 監視できる条件は、結ぶ線分の全体が  $N$  角形の内部または周上に含まれていること
- 見張りは最小何人必要か？



# Sample Input 1



# Sample Input 1



OK!

# 解法

- 幾何パート
  - 「頂点  $A$  から頂点  $B$  を監視できるか？」を各頂点の組に対して調べる
    - 線分が多角形の内部または周上に含まれているか
- グラフパート
  - 頂点を最小いくつ選べば見張れるか求める
    - 「最小支配集合」(minimum dominating set)

# 幾何パート

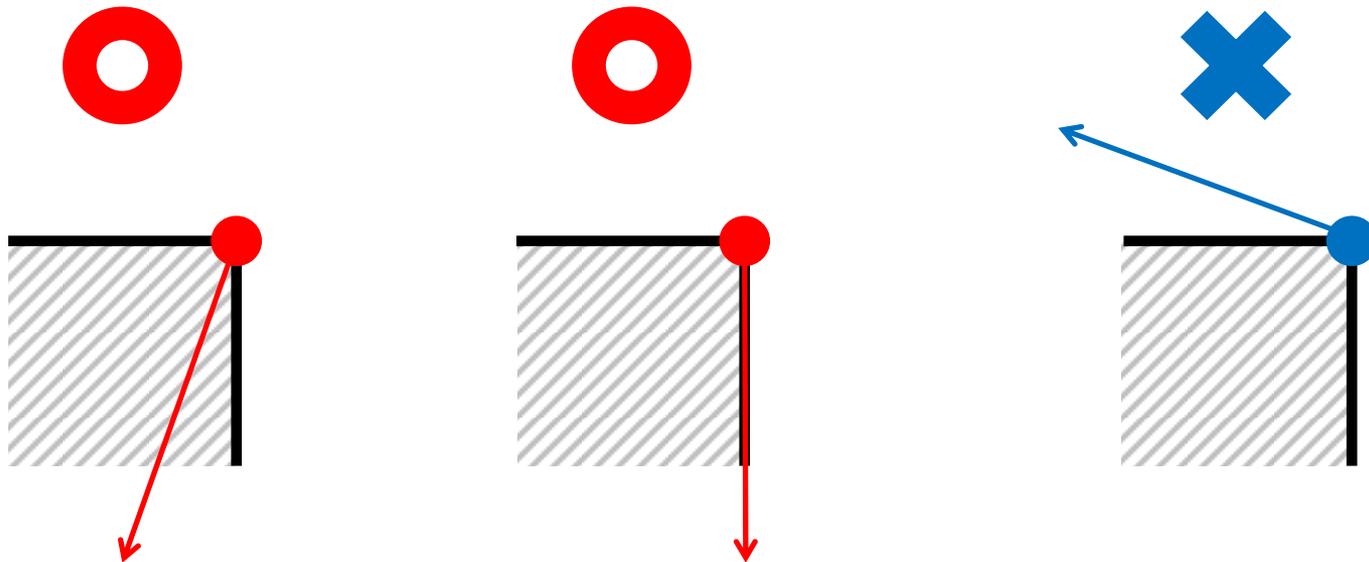
- がんばる
- 適宜ライブラリを頼りにしましょう
- 混乱せず書ける方針の 1 つを紹介します

# 幾何パート

- 線分が多角形の内部または周上にあるための条件
  1. 線分の端 ( : 多角形の頂点) からちょっと進んだ部分が多角形の外部に出ない
  2. 途中で多角形の外部に出ない

# 幾何パート

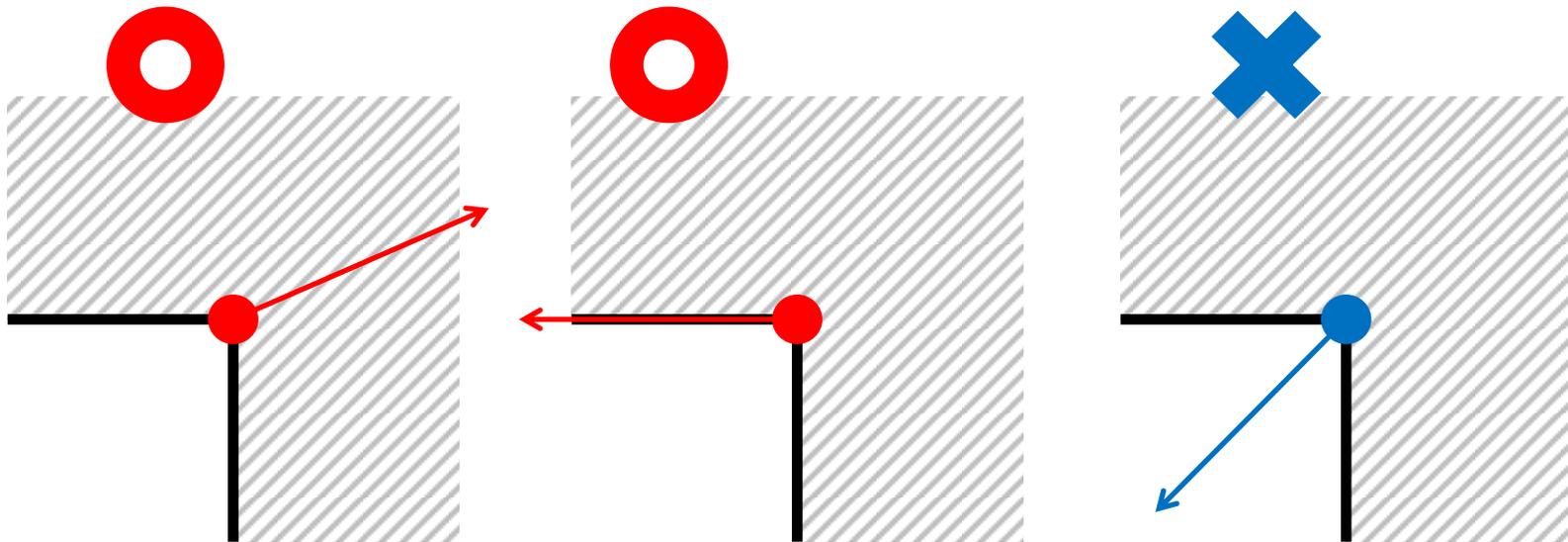
1. 線分の端（：多角形の頂点）からちょっと進んだ部分が多角形の外部に出ない  
– 角の内部方向に進んでいるかどうか



(斜線の側が内部)

# 幾何パート

1. 線分の端（：多角形の頂点）からちょっと進んだ部分が多角形の外部に出ない  
– 角の内部方向に進んでいるかどうか



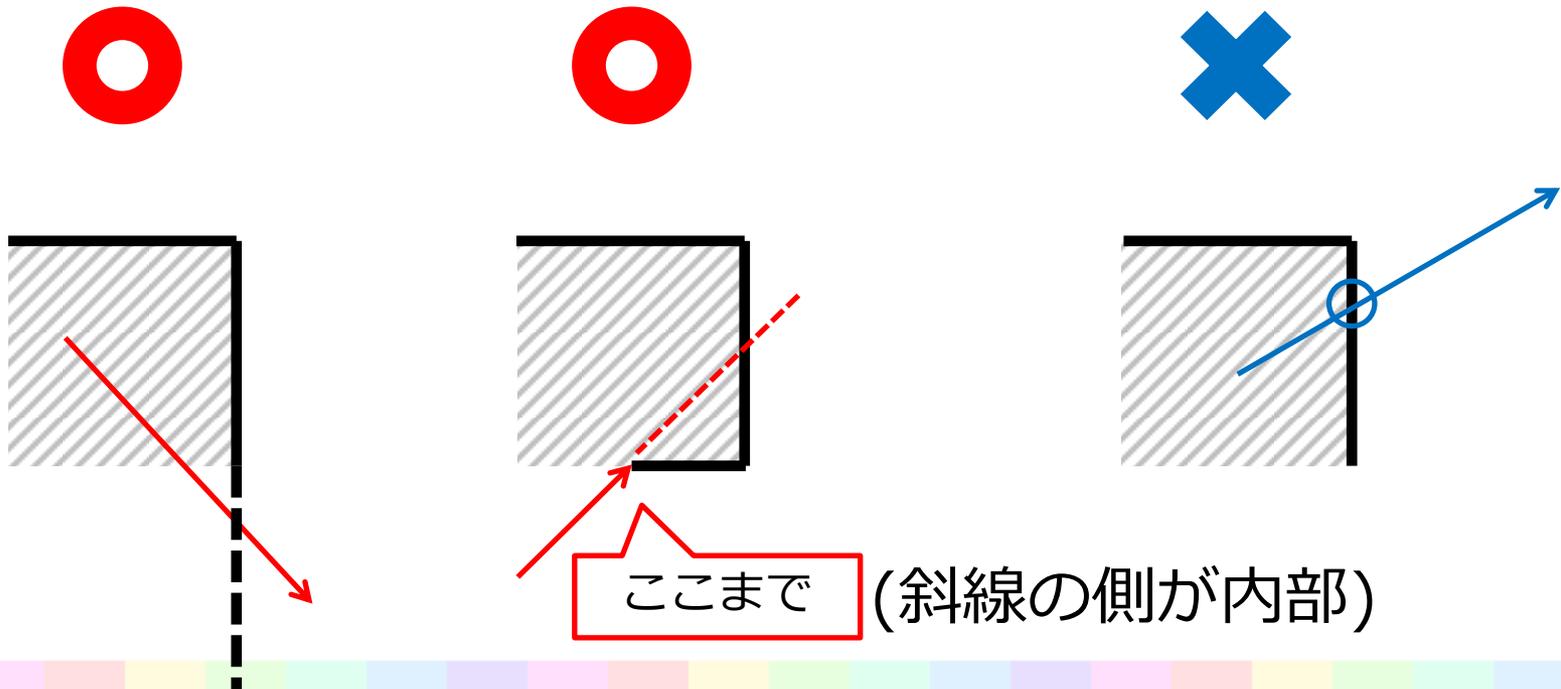
(斜線の側が内部)

# 幾何パート

## 2. 途中で多角形の外部に出ない

– 途中で**辺**を越えて外部に出ない

- 線分の交差判定 (端点以外で交わっていないか)

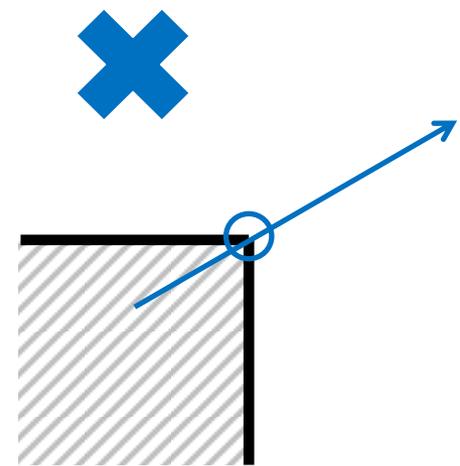
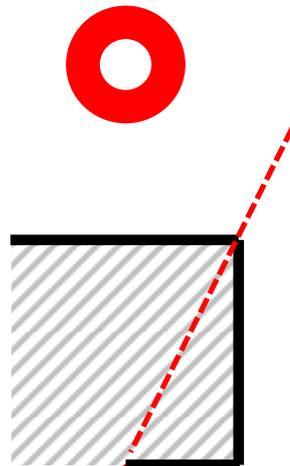
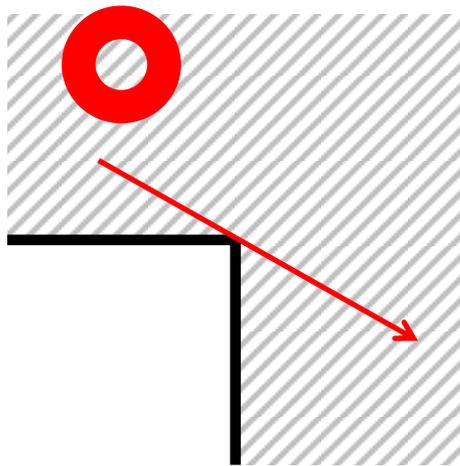


# 幾何パート

## 2. 途中で多角形の外部に出ない

– 途中で**頂点**を越えて外部に出ない

- 頂点が線分上か／双方向について角の内部判定



ここまで (斜線の側が内部)

# 最近流行の便利関数

- 原点を始点とするベクトル  $\vec{a}$ ,  $\vec{b}$ ,  $\vec{c}$  について,  $\vec{c}$  が「 $\vec{a}$  から  $\vec{b}$  へ反時計回りで回る角」の内部・辺上・外部のどこにあるか  
– どの場合にどういう値になるか (場合分けでわかる) を考えて動作を理解しましょう

$$\text{sign}(\vec{a} \times \vec{c}) - \text{sign}(\vec{b} \times \vec{c}) - \text{sign}(\vec{a} \times \vec{b})$$

sign は負なら -1, 正なら +1, 0 なら 0

$\vec{a} = (x_a, y_a), \vec{b} = (x_b, y_b)$  に対して  $\vec{a} \times \vec{b} = x_a y_b - x_b y_a$

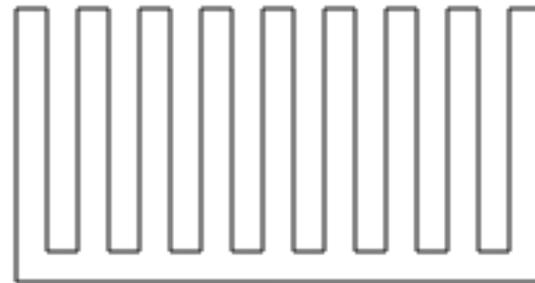
# グラフパート

- 頂点の集合すべてについて, それらですべての頂点を見張れるか調べる?
  - $2^N$  通り (遅い)
- 何かしらの工夫が必要

# ポイント

- 答えは実はあまり大きくなならない
- 各辺が座標軸に平行な多角形に対し, 必要な見張りの人数は  $[N/4]$  以下 (Kahn, Klawe, Kleitman, 1980)
  - 証明は結構大変 (四角形分割できるのを示す)
  - 一般の多角形だと  $[N/3]$  以下 (三角形分割)

$[N/4]$  人必要な例 →



# 工夫の例

- 8 頂点以下の集合をすべて調べる
  - 無理だったら答えは 9
  - ${}_{38}C_0 + {}_{38}C_1 + \dots + {}_{38}C_8 = 64,869,364$ 
    - 判定にビット演算を用いる, などうまくやれば間に合う
    - $N = 40$  ( $\rightarrow$  サイズ 9 まで) でもなんとか間に合うようにできます

# 工夫の例

- 反復深化深さ優先探索 (iterative deepening depth-first search) を用いる
  - 答えの上限を気にせず書ける

# 工夫の例

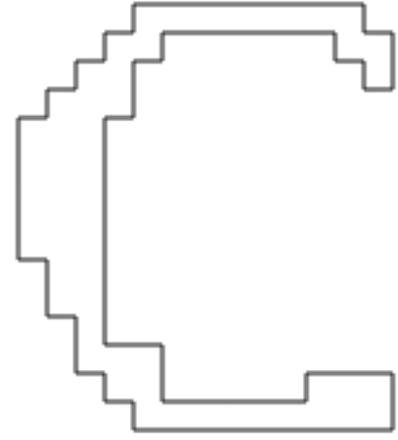
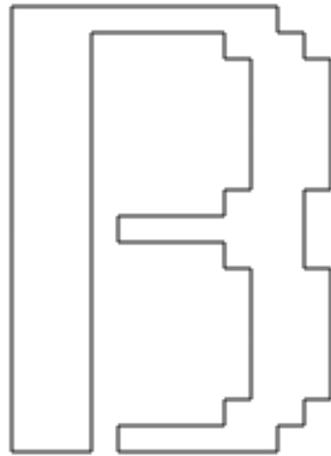
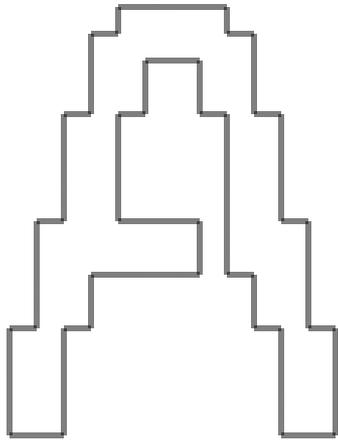
- 深さ優先探索 + 枝刈り
  - 今選んでいる頂点たちですべて見張れていたらバックトラック, だったり
  - 新たに頂点を選んでも見張れる頂点が増えないなら選ばない, だったり
  - 今まで見つかった答えのサイズ以上になったらバックトラック, だったり
    - 答えが小さいから有効

# 工夫の例

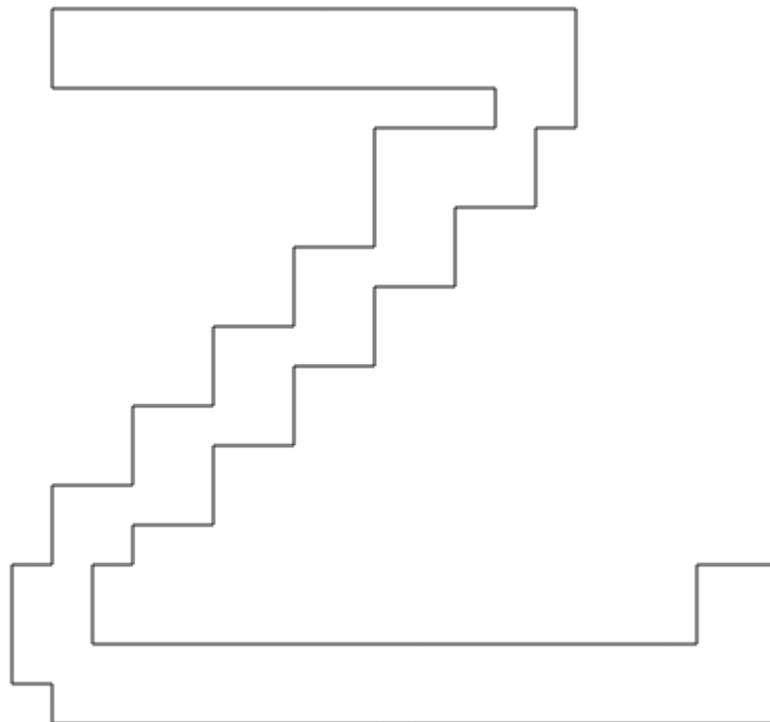
- 一般の最小支配集合問題 (NP 困難) に対する指数時間アルゴリズムを用いる
  - $O(1.4969^N)$  時間とかあるらしいです
  - この問題で使うのはやや大げさかもしれませんが、ライブラリ化してあるなら十分あり

# ジャッジデータ紹介

- この子たち 26 人でいろいろな誤答を捕らえてくれました



# ジャッジデータ紹介



# 結果

- Accepted / Trying Teams / Submission
  - 10 / 11 / 36
- First Accepted
  - Operasan (in 海ほたる) (141:41)