Sightseeing in Kyoto

Editorial: Hirotaka Yoneda (square1001)

[[Front cover taken on Nov 23rd, 2021]]

京都市は有名な観光地ですが…

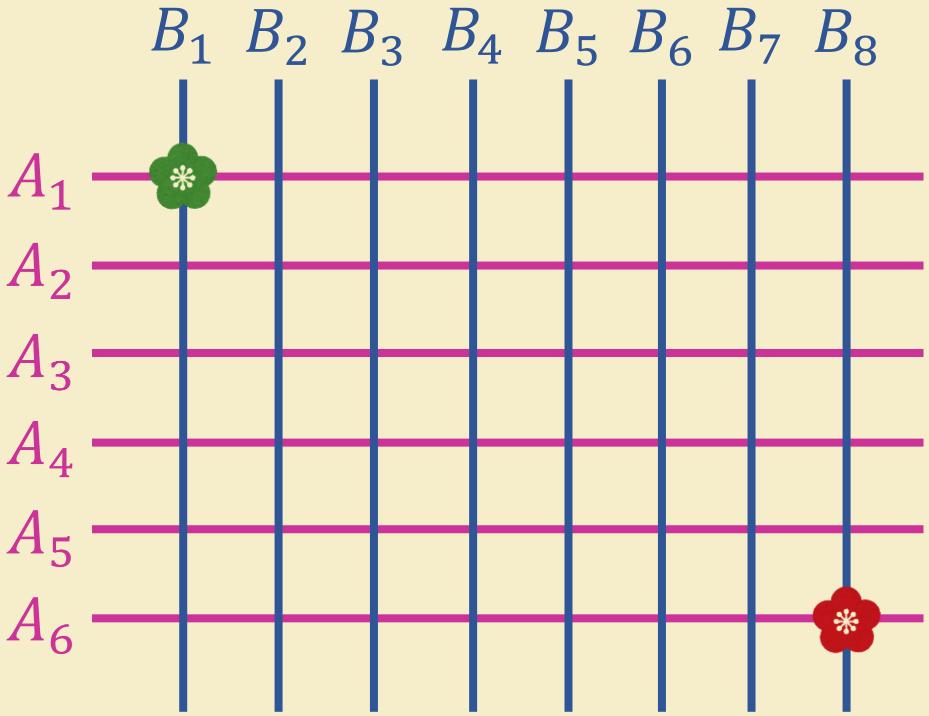
京都市は「碁盤目状の道路」でも有名です

問題概要

横方向の道路が H 本、縦方向の道路が W 本あり
右図のように「碁盤目状」になっています

道路によって歩く速度が異なってきます

- 北から i 本目の横方向の道路は
  1 マス分歩くのに A_i 秒かかる
- 西から j 本目の横方向の道路は
  1 マス分歩くのに B_j 秒かかる



左上端 → 右下端 には最短で何秒かかる？

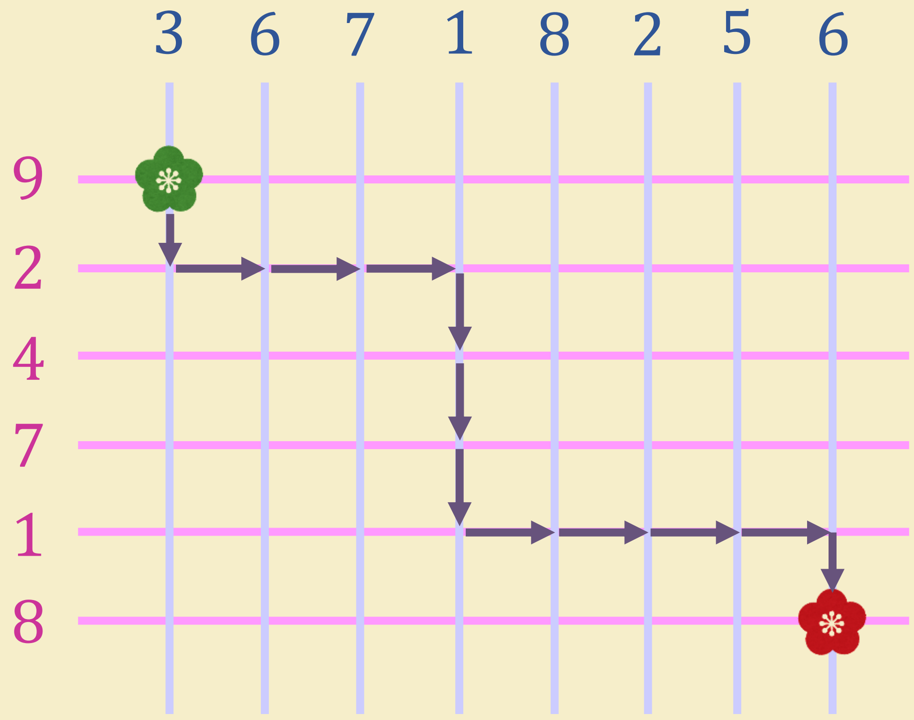
ただし、遠回りをしてはならないものとします（つまり、右方向 or 下方向にしか動けない）

問題概要

右図は $H = 6, W = 8$ の例

このように進むのが「最短」になります

移動時間：3+2+2+2+1+1+1+1+1+1+1+6 = 22

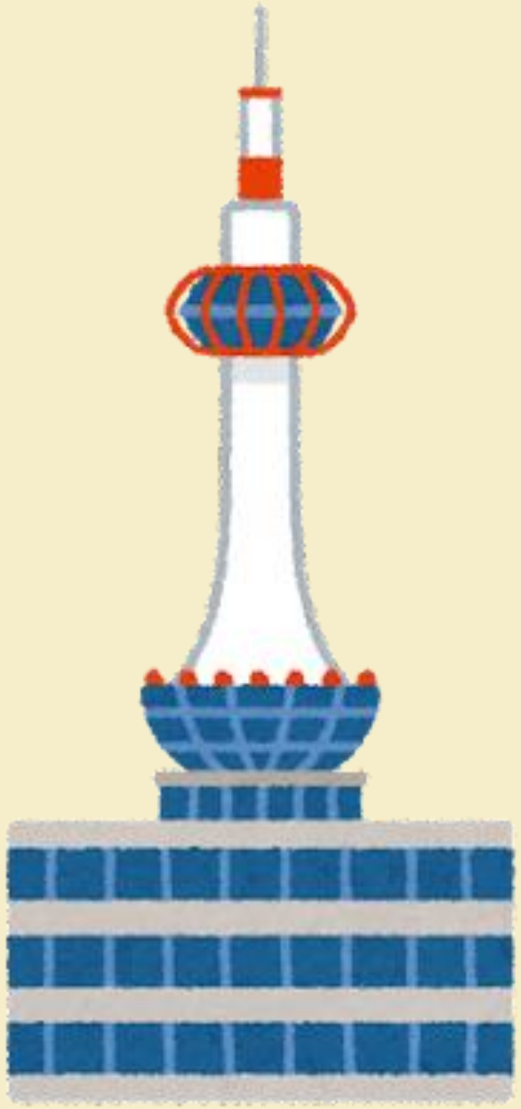


制約

- 小課題 1（10 点）：$H, W \leq 10^3$
- 小課題 2（30 点）：$H, W \leq 10^5; A_i, B_j \leq 10^3$
- 小課題 3（60 点）：$H, W \leq 10^5$

小課題１（10点）

制約：$H, W \leq 1000$



小課題 I — $H, W \leq 1000$

動的計画法 (DP) で解くことを考えます

$d[i][j]$ =「交差点 (i, j) に到達するための最短の時間」を順々に求めていきます

小課題 I — $H, W \leq 1000$

動的計画法 (DP) で解くことを考えます

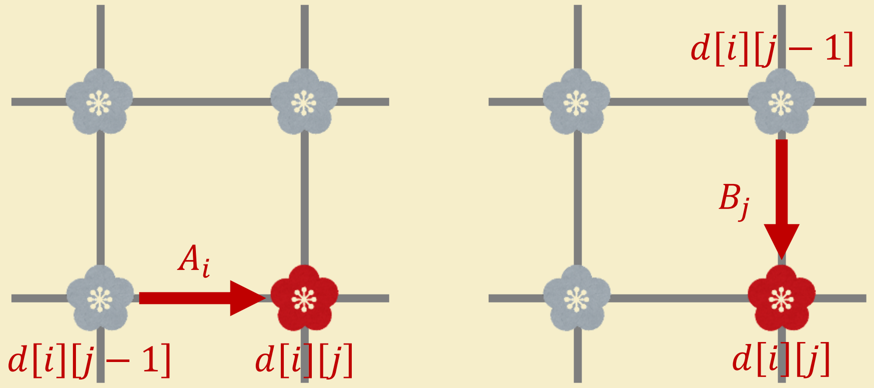
$d[i][j]$ =「交差点 (i, j) に到達するための最短の時間」を順々に求めていきます

$d[i][j]$ を求めたい

交差点 (i, j) に到達する方法は右図の 2 通りに分けられます

- 左から進む場合：最短距離 $d[i][j-1] + A_i$
- 上から進む場合：最短距離 $d[i-1][j] + B_j$

よって $d[i][j] = \min\left(d[i][j-1] + A_i, d[i-1][j] + B_j\right)$



小課題 I — $H, W \leq 1000$

この式にしたがって、左上の交差点から順に求めます

答えは $d[H][W]$ になります

計算量は $O(HW)$ → 小課題 I には通ります

```
#include <iostream>
#include <algorithm>
using namespace std;

const long long INF = (1LL << 62);
int H, W, A[1009], B[1009]; long long d[1009][1009];

int main() {
    cin >> H >> W;
    for (int i = 1; i <= H; i++) cin >> A[i];
    for (int i = 1; i <= W; i++) cin >> B[i];
    for (int i = 1; i <= H; i++) {
        for (int j = 1; j <= W; j++) {
            if (i == 1 && j == 1) continue;
            long long v1 = (j>=2 ? d[i][j-1] + A[i] : INF);
            long long v2 = (i>=2 ? d[i-1][j] + B[j] : INF);
            d[i][j] = min(v1, v2);
        }
    }
    cout << d[H][W] << endl;
}
```

小課題 2（30 点）

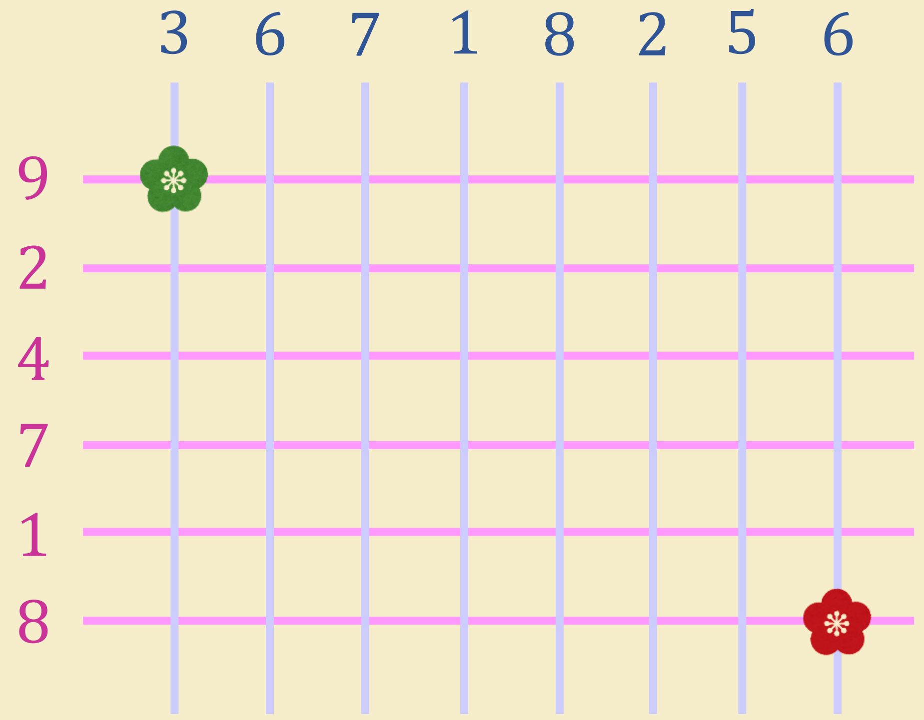
制約：$H, W \leq 10^5; A_i, B_j \leq 1000$

小課題 2 — $A_i, B_j \leq 1000$

右図の例を考えてみよう

どのような道路を通ることになるか？

逆に、どのような道路は確実に通らないか？



小課題 2 — $A_i, B_j \leq 1000$

右図の例を考えてみよう

どのような道路を通ることになるか？

逆に、どのような道路は確実に通らないか？

実は、× を書いた道路は「確実に通らない」！

小課題 2 — $A_i, B_j \leq 1000$

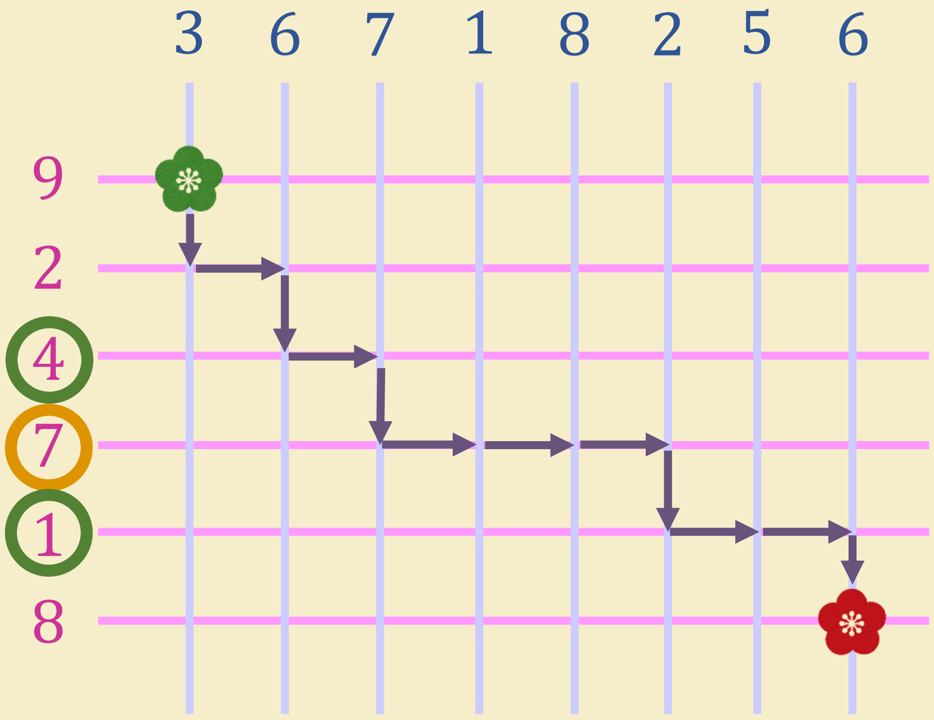
重要な性質

以下の道路は絶対に通らない

- $A_{i-1} \leq A_i \geq A_{i+1}$ となるような、横方向の道路 i
- $B_{i-1} \leq B_i \geq B_{i+1}$ となるような、縦方向の道路 j

具体例

仮に右図の「7」の道路を通る経路が最適だとする

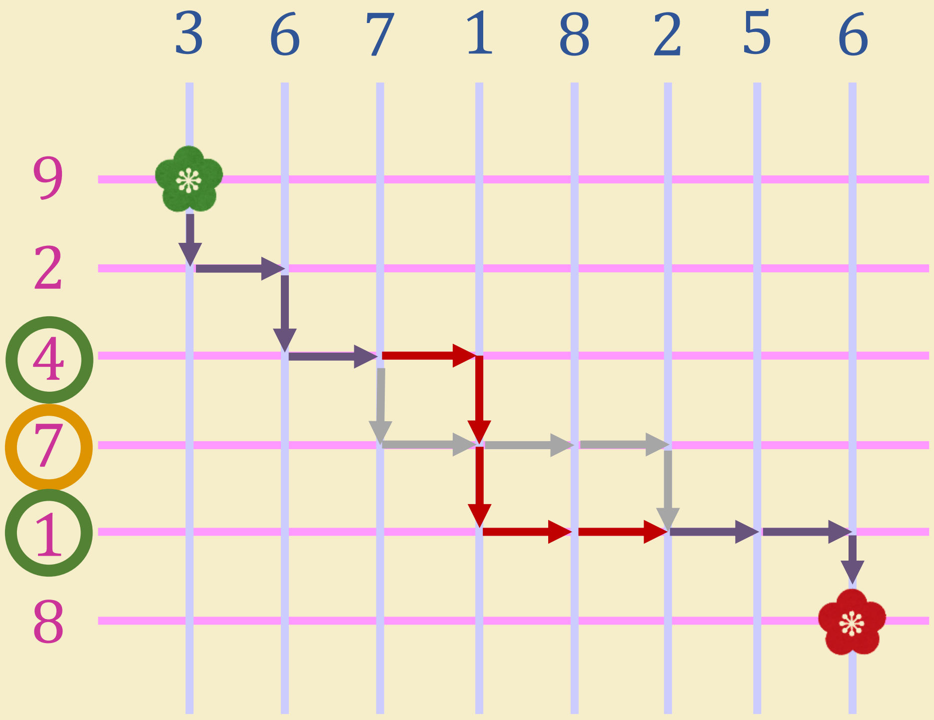


小課題 2 — $A_i, B_j \leq 1000$

重要な性質

以下の道路は絶対に通らない

- $A_{i-1} \leq A_i \geq A_{i+1}$ となるような、横方向の道路 i
- $B_{i-1} \leq B_i \geq B_{i+1}$ となるような、縦方向の道路 j



具体例

仮に右図の「7」の道路を通る経路が最適だとする

これは、赤い（縦の「1」を通る）経路に変えた方が明らかに得

（理由は、縦の部分 [7+2 > 1+1] と横の部分 [7+7+7 > 4+1+1] の両方で、全区間で短縮されるから）

小課題 2 — $A_i, B_j \leq 1000$

重要な性質

以下の道路は絶対に通らない

- $A_{i-1} \leq A_i \geq A_{i+1}$ となるような、横方向の道路 i
- $B_{i-1} \leq B_i \geq B_{i+1}$ となるような、縦方向の道路 j

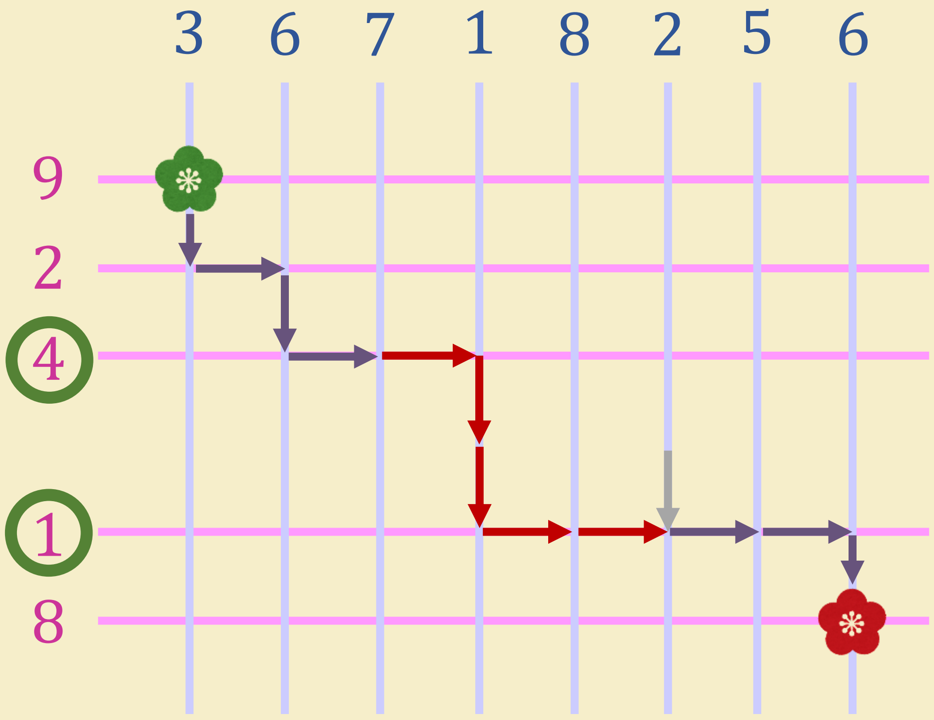
証明

A_i の道路を交差点 $(i, l) \to \cdots \to (i, r)$ で通るとする

ここで、$B_l, B_{l+1}, \ldots, B_r$ の最小値を B_k とする

すると、$(i-1, l) \to \cdots \to (i-1, k) \to (i, k) \to (i+1, k) \to \cdots \to (i+1, r)$ の経路にした方が絶対に得

だから、A_i の道路は "存在意義がない"（B_j の道路についても同様）



小課題 2 — $A_i, B_j \leq 1000$

重要な性質

以下の道路は絶対に通らない

- $A_{i-1} \leq A_i \geq A_{i+1}$ となるような、横方向の道路 i
- $B_{i-1} \leq B_i \geq B_{i+1}$ となるような、縦方向の道路 j

この要領で道路を消していくとどうなる…？

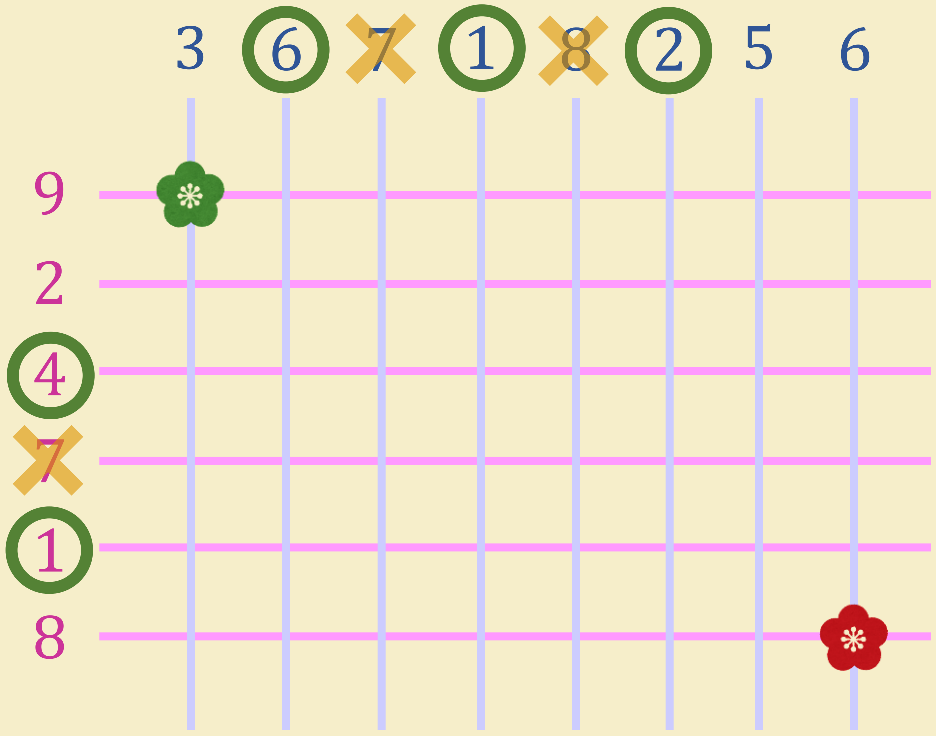
小課題 2 — $A_i, B_j \leq 1000$

重要な性質

以下の道路は絶対に通らない

- $A_{i-1} \leq A_i \geq A_{i+1}$ となるような、横方向の道路 i
- $B_{i-1} \leq B_i \geq B_{i+1}$ となるような、縦方向の道路 j

この要領で道路を消していくとどうなる…？



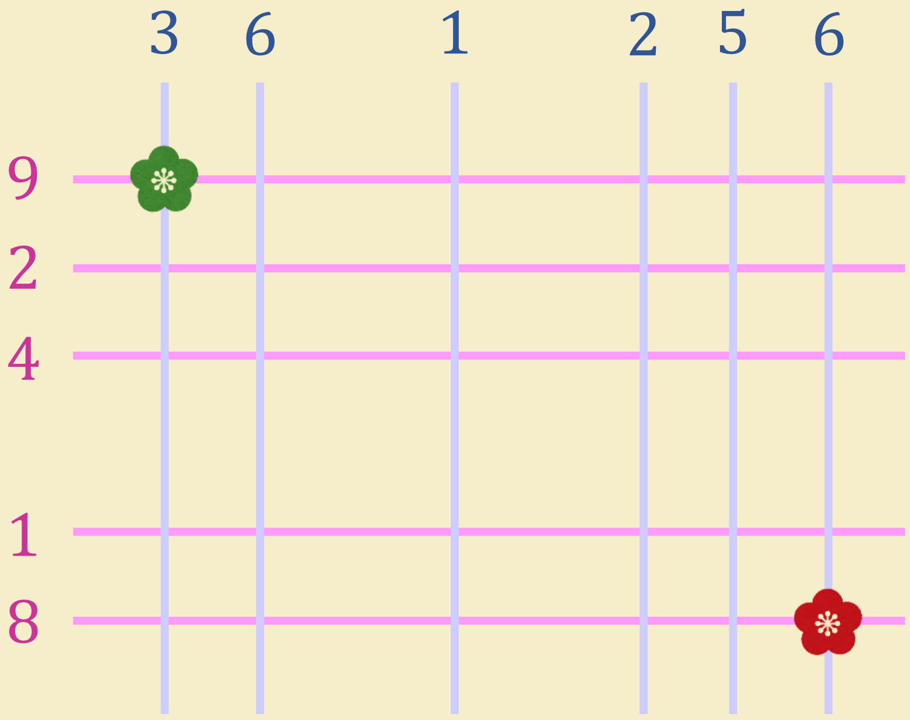
小課題 2 — $A_i, B_j \leq 1000$

重要な性質

以下の道路は絶対に通らない

- $A_{i-1} \leq A_i \geq A_{i+1}$ となるような、横方向の道路 i
- $B_{i-1} \leq B_i \geq B_{i+1}$ となるような、縦方向の道路 j

この要領で道路を消していくとどうなる…？



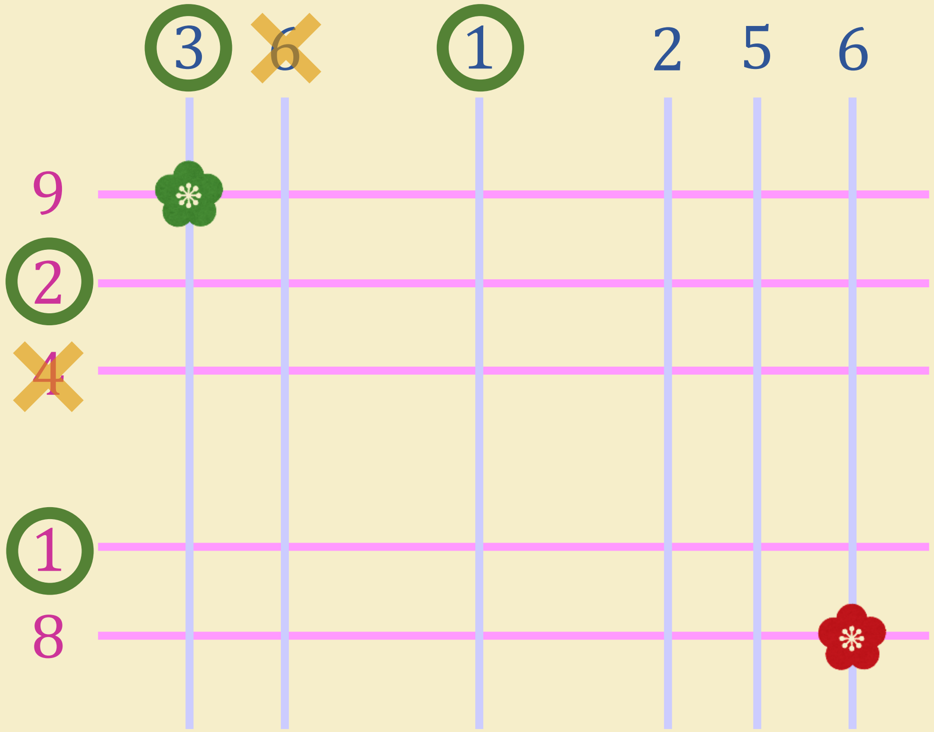
小課題 2 — $A_i, B_j \leq 1000$

重要な性質

以下の道路は絶対に通らない

- $A_{i-1} \leq A_i \geq A_{i+1}$ となるような、横方向の道路 i
- $B_{i-1} \leq B_i \geq B_{i+1}$ となるような、縦方向の道路 j

この要領で道路を消していくとどうなる…？



小課題 2 — $A_i, B_j \leq 1000$

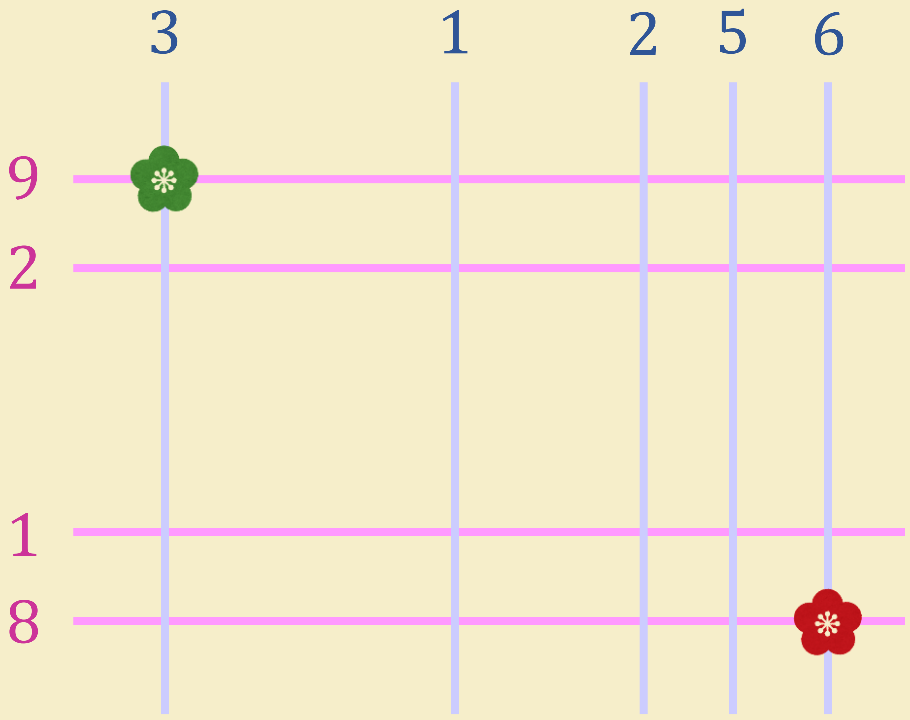
重要な性質

以下の道路は絶対に通らない

- $A_{i-1} \leq A_i \geq A_{i+1}$ となるような、横方向の道路 i
- $B_{i-1} \leq B_i \geq B_{i+1}$ となるような、縦方向の道路 j

この要領で道路を消していくとどうなる…？

→ たくさんの道路が消えた！



小課題 2 — $A_i, B_j \leq 1000$

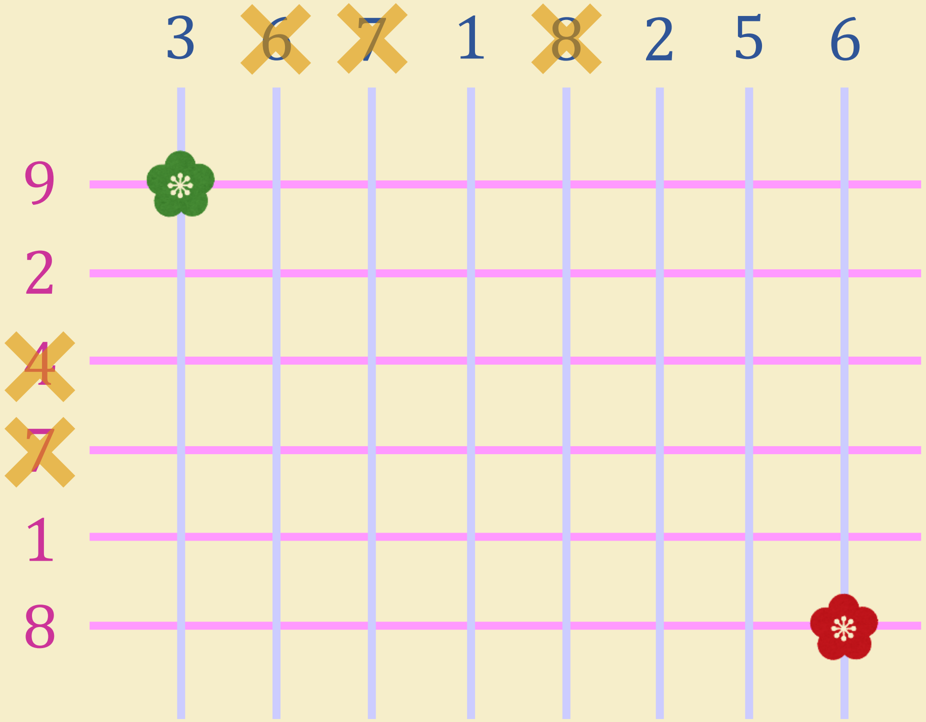
結局、残る道路は…？

- A_i の左から or 右からの最小値を更新する道路
- B_j の左から or 右からの最小値を更新する道路

$A_i, B_j \leq 1000$ の場合…

残る道路は縦横それぞれ 2000 本以下

→ これなら DP で最短経路が求まる！



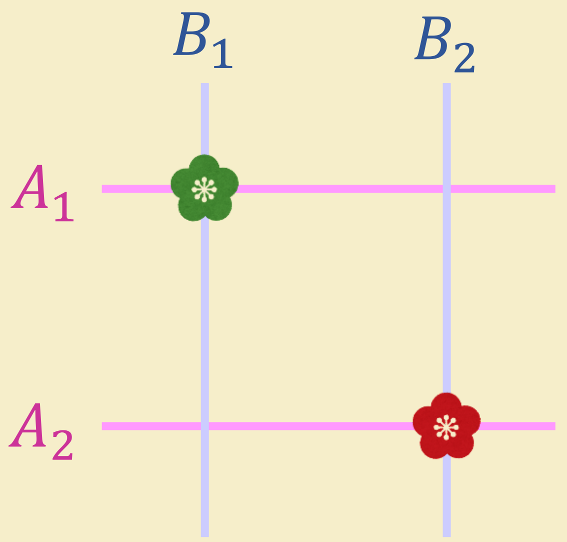
全体計算量は $O(MAX^2)$ [ただし MAX は A_i, B_j の最大値]

満点解法

制約：$H, W \leq 10^5$

小課題 3 — $H, W \leq 100000$

ところで、$H = 2, W = 2$ の場合を考えよう



$H = 2, W = 2$ の場合

経路は 2 つあるが、どちらを通った方がよいか？

- $(1,1) \to (1,2) \to (2,2)$： $A_1 + B_2$ 秒
- $(1,1) \to (2,1) \to (2,2)$： $B_1 + A_2$ 秒

$A_1 + B_2 < B_1 + A_2 \Leftrightarrow A_2 - A_1 > B_2 - B_1$ のとき前者の方が得

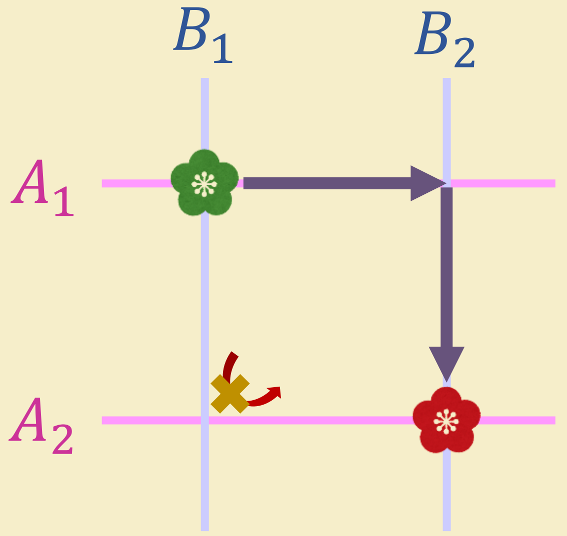
小課題 3 — $H, W \leq 100000$

ところで、$H = 2, W = 2$ の場合を考えよう

$H = 2, W = 2$ の場合

経路は 2 つあるが、どちらを通った方がよいか？

- $(1,1) \rightarrow (1,2) \rightarrow (2,2)$： $A_1 + B_2$ 秒
- $(1,1) \rightarrow (2,1) \rightarrow (2,2)$： $B_1 + A_2$ 秒



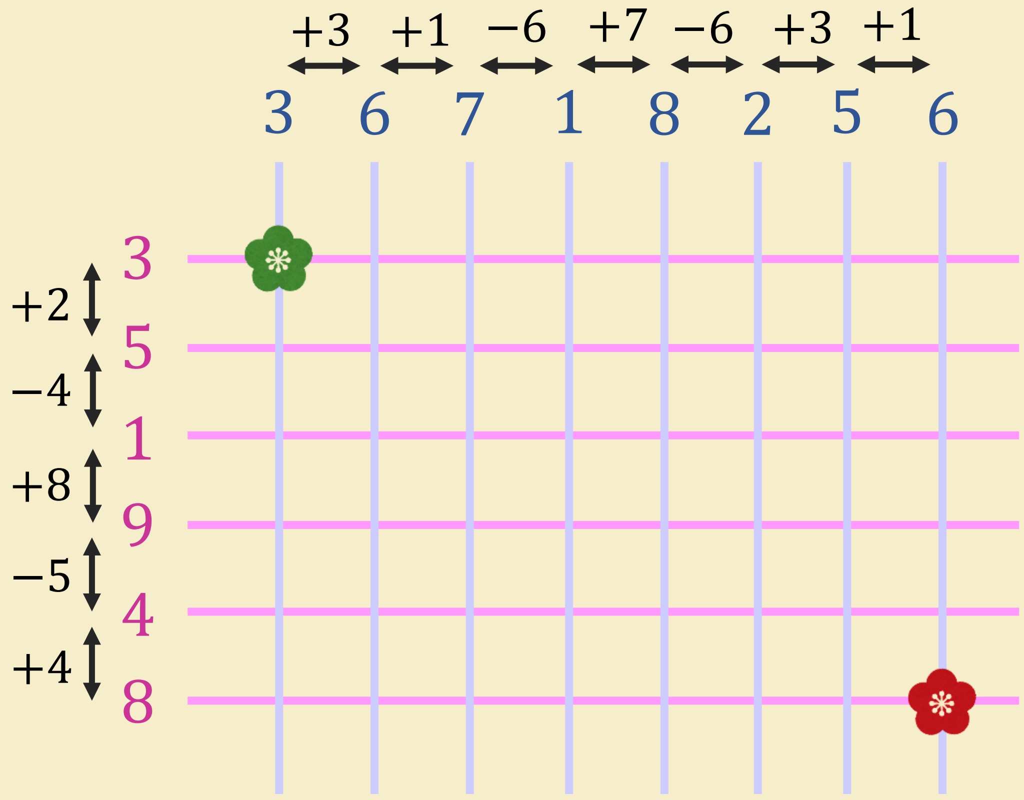
$A_1 + B_2 < B_1 + A_2 \Leftrightarrow A_2 - A_1 > B_2 - B_1$ のとき前者の方が得

つまり、交差点 $(2,1)$ で左向きに曲がることは許されない！

小課題 3 — $H, W \leq 100000$

これを、一般の場合に適用してみましょう

（右図の黒い数は「差分」を表しています）



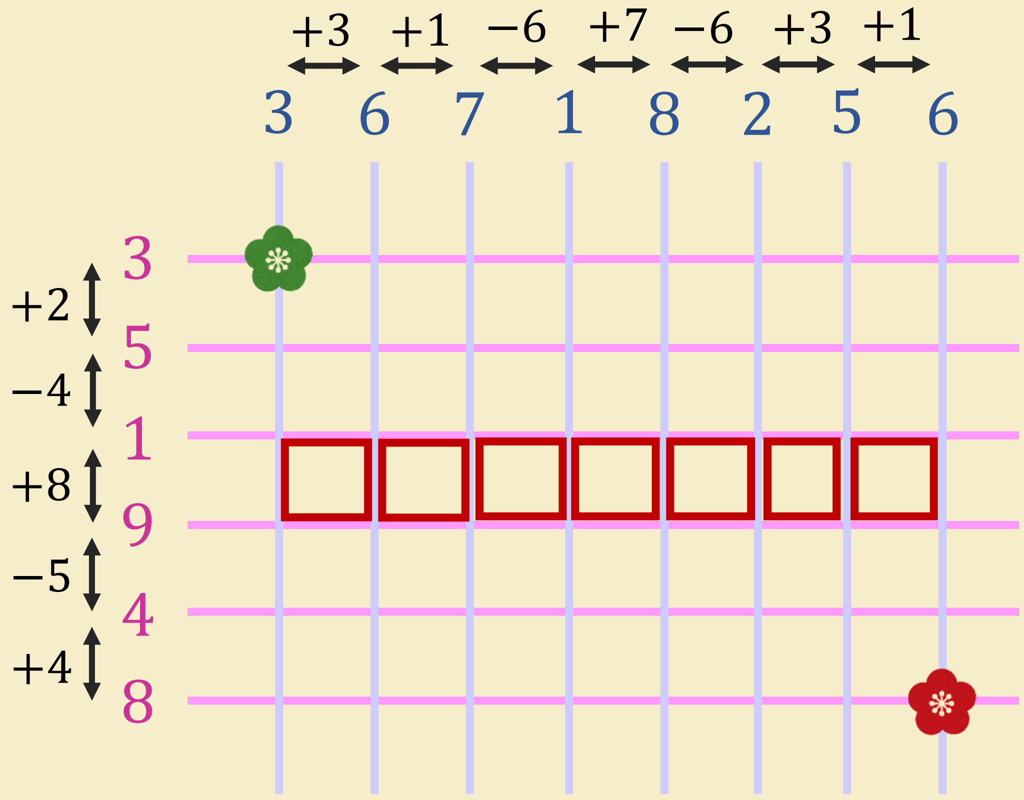
小課題 3 — $H, W \leq 100000$

これを、一般の場合に適用してみましょう

（右図の黒い数は「差分」を表しています）

「+8」は全体で最も大きい差分なので

この行のすべての四角に条件を適用させると…



小課題 3 — $H, W \leq 100000$

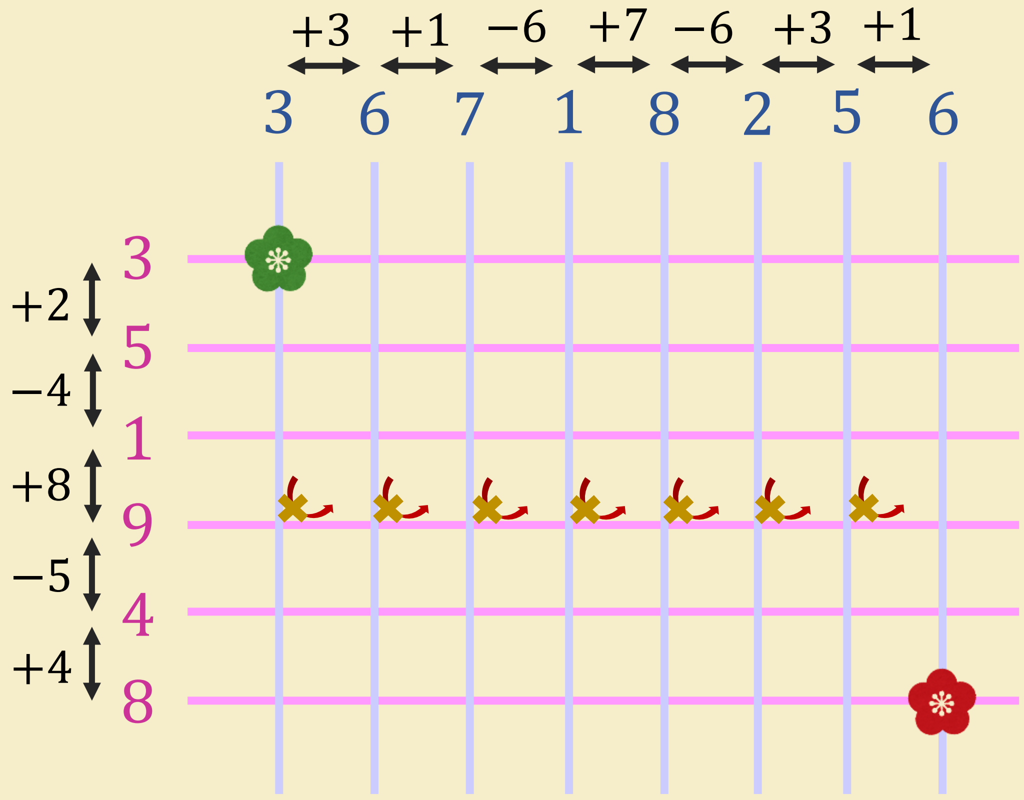
これを、一般の場合に適用してみましょう

（右図の黒い数は「差分」を表しています）

「+8」は全体で最も大きい差分なので

この行のすべての四角に条件を適用させると…

“左に曲がることはできない”



小課題 3 — $H, W \leq 100000$

これを、一般の場合に適用してみましょう

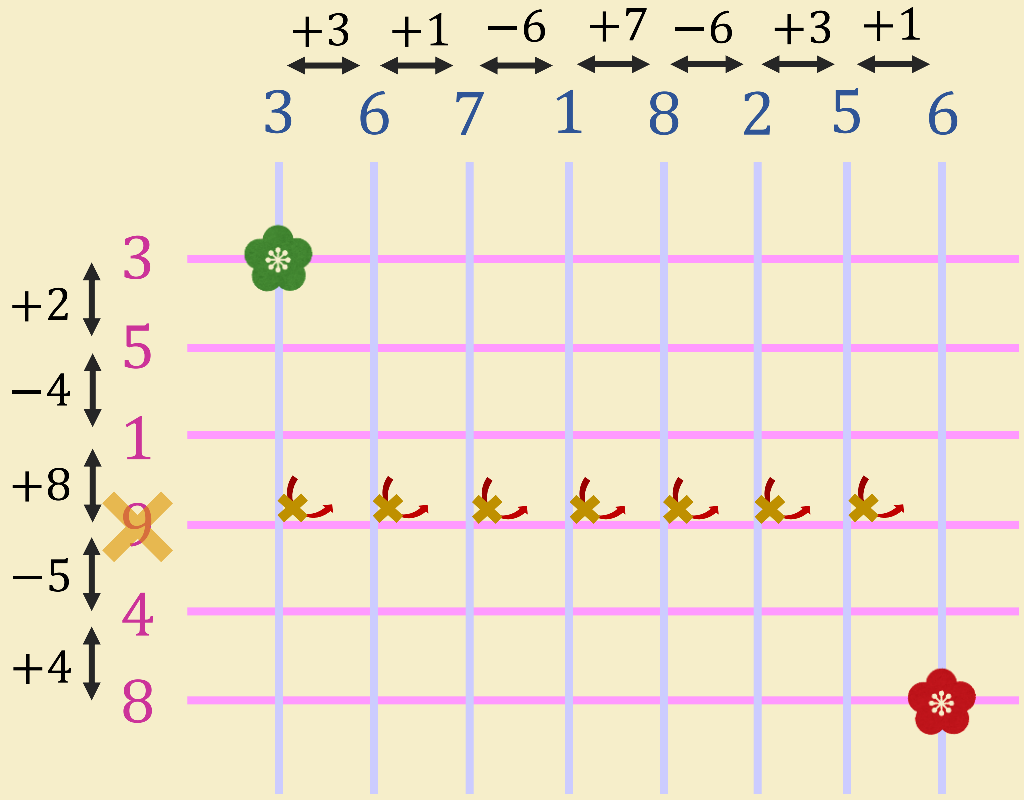
（右図の黒い数は「差分」を表しています）

「+8」は全体で最も大きい差分なので

この行のすべての四角に条件を適用させると…

“左に曲がることはできない”

つまり、「9」の道路は絶対に通ることができない！



小課題 3 — $H, W \leq 100000$

これを、一般の場合に適用してみましょう

（右図の黒い数は「差分」を表しています）

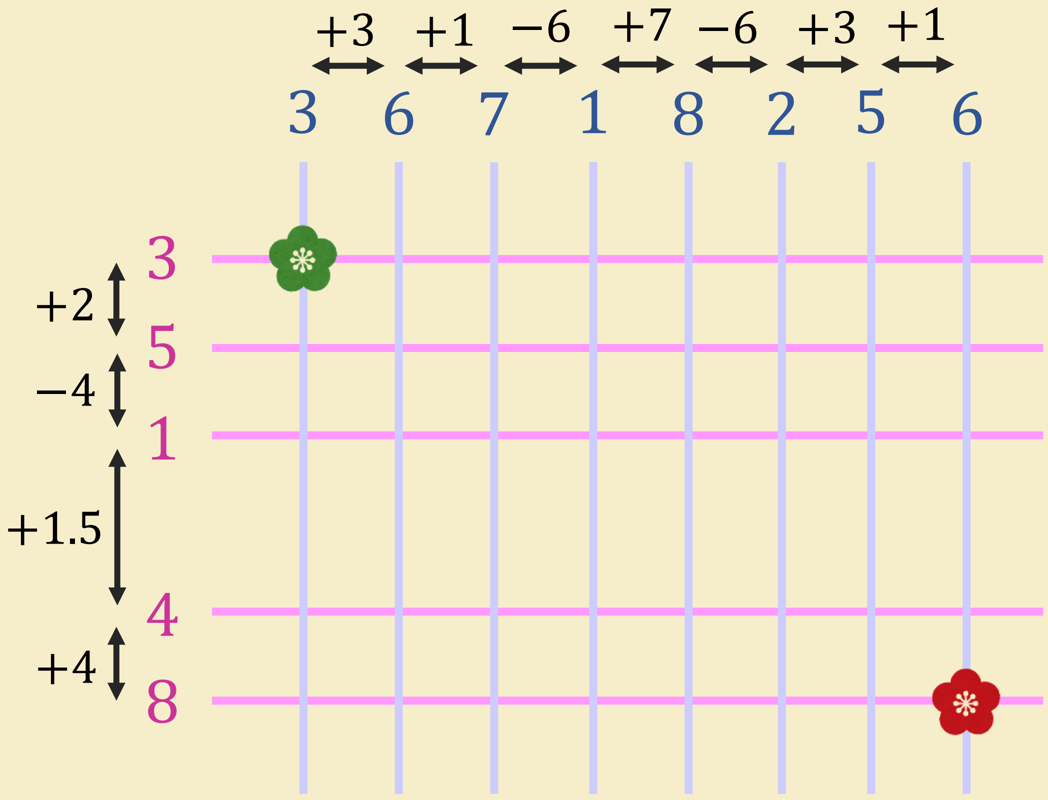
「+8」は全体で最も大きい差分なので

この行のすべての四角に条件を適用させると…

“左に曲がることはできない”

つまり、「9」の道路は絶対に通ることができない！

これで 1 つ道路が消えた



小課題 3 — $H, W \leq 100000$

これを、一般の場合に適用してみましょう

（右図の黒い数は「差分」を表しています）

「+8」は全体で最も大きい差分なので

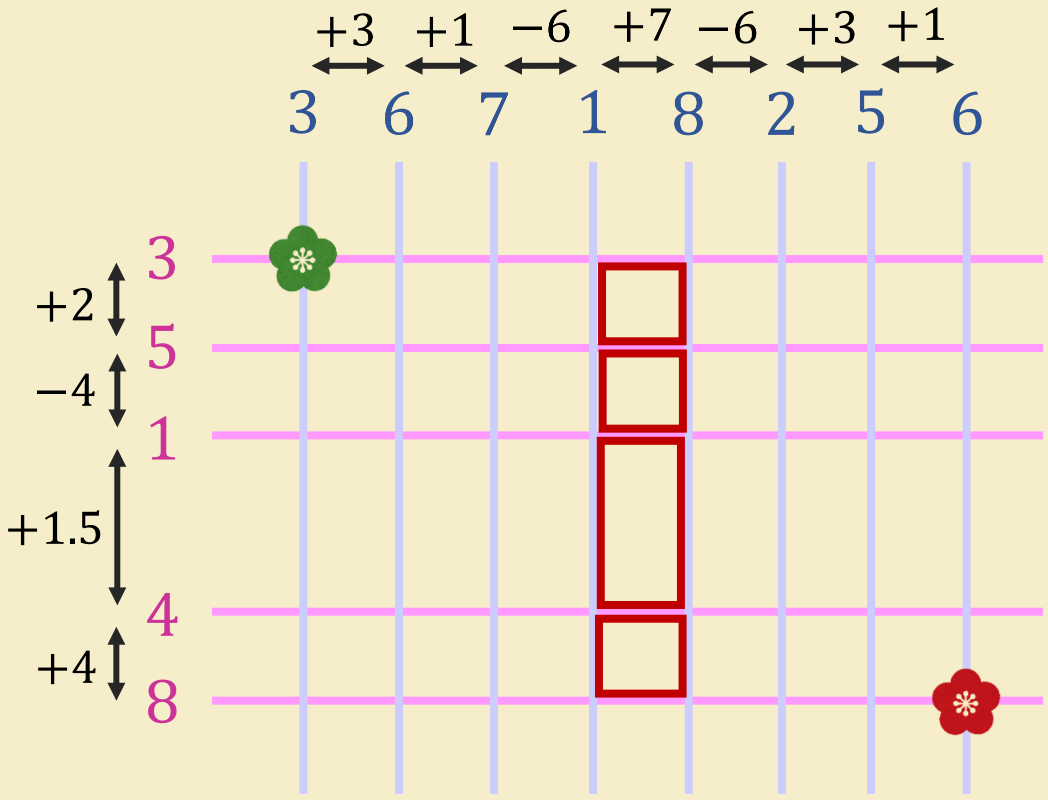
この行のすべての四角に条件を適用させると…

“左に曲がることはできない”

つまり、「9」の道路は絶対に通ることができない！

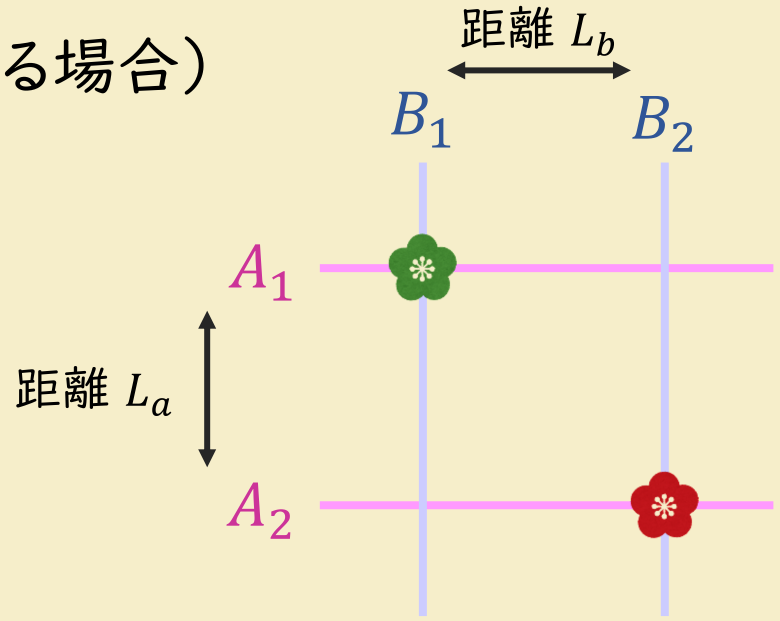
これで 1 つ道路が消えた

次に大きい「+7」に条件を適用させたい



小課題 3 — $H, W \leq 100000$

ここで、$H = 2, W = 2$ に戻って考えよう（道路が離れている場合）



$H = 2, W = 2$ の場合

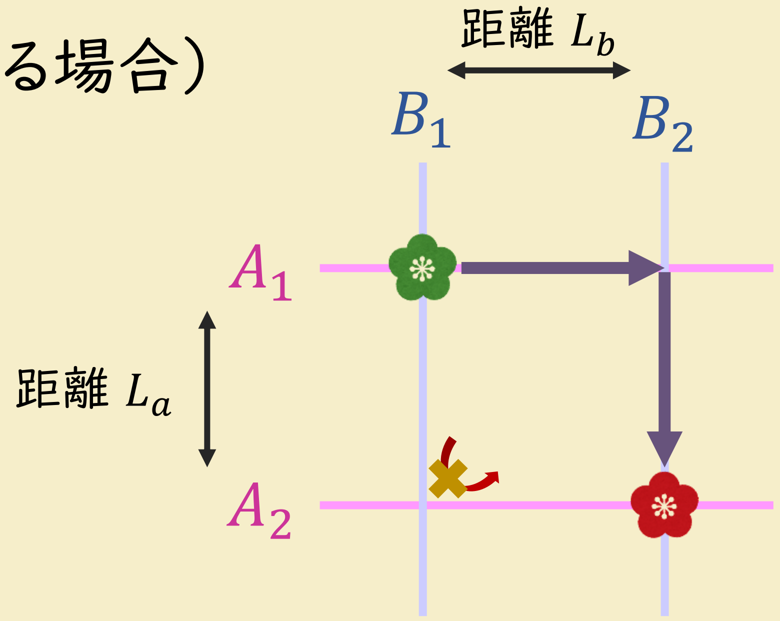
経路は 2 つあるが、どちらを通った方がよいか？

- $(1,1) \to (1,2) \to (2,2)$： $A_1 \cdot L_b + B_2 \cdot L_a$ 秒
- $(1,1) \to (2,1) \to (2,2)$： $B_1 \cdot L_a + A_2 \cdot L_b$ 秒

$A_1 \cdot L_b + B_2 \cdot L_a < B_1 \cdot L_a + A_2 \cdot L_b \Leftrightarrow \frac{A_2 - A_1}{L_a} > \frac{B_2 - B_1}{L_b}$ のとき前者の方が得

小課題 3 — $H, W \leq 100000$

ここで、$H = 2, W = 2$ に戻って考えよう（道路が離れている場合）



$H = 2, W = 2$ の場合

経路は 2 つあるが、どちらを通った方がよいか？

- $(1,1) \to (1,2) \to (2,2)$： $A_1 \cdot L_b + B_2 \cdot L_a$ 秒
- $(1,1) \to (2,1) \to (2,2)$： $B_1 \cdot L_a + A_2 \cdot L_b$ 秒

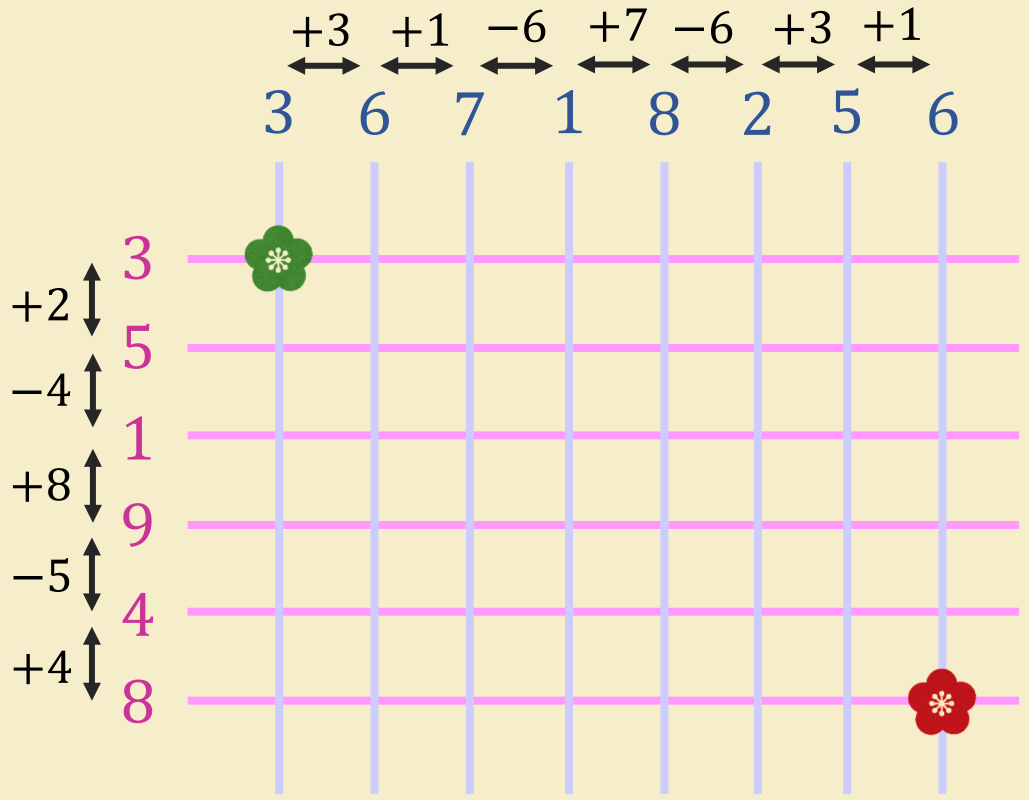
$A_1 \cdot L_b + B_2 \cdot L_a < B_1 \cdot L_a + A_2 \cdot L_b \Leftrightarrow \frac{A_2 - A_1}{L_a} > \frac{B_2 - B_1}{L_b}$ のとき前者の方が得

→ つまり、（差分）÷（距離）で運命が分かれる！（これを“傾き”ということにする）

小課題 3 — $H, W \leq 100000$

最初からやってみよう！

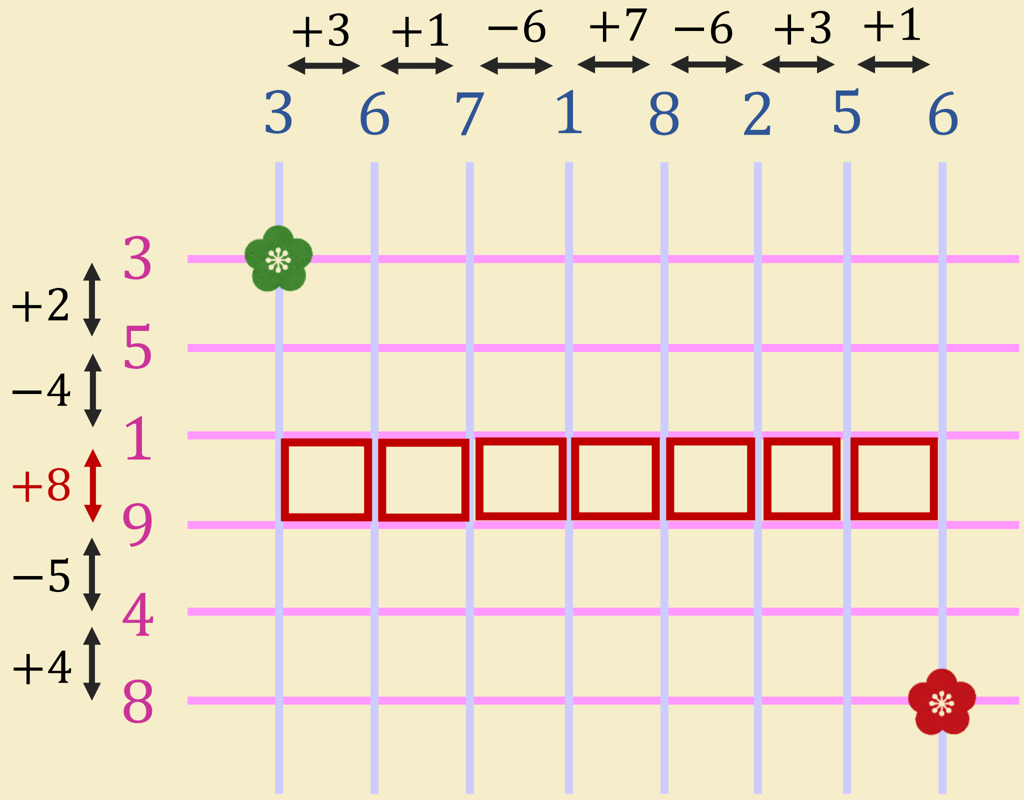
（右図の黒い数は「傾き」を表しています）



小課題 3 — $H, W \leq 100000$

最初からやってみよう！

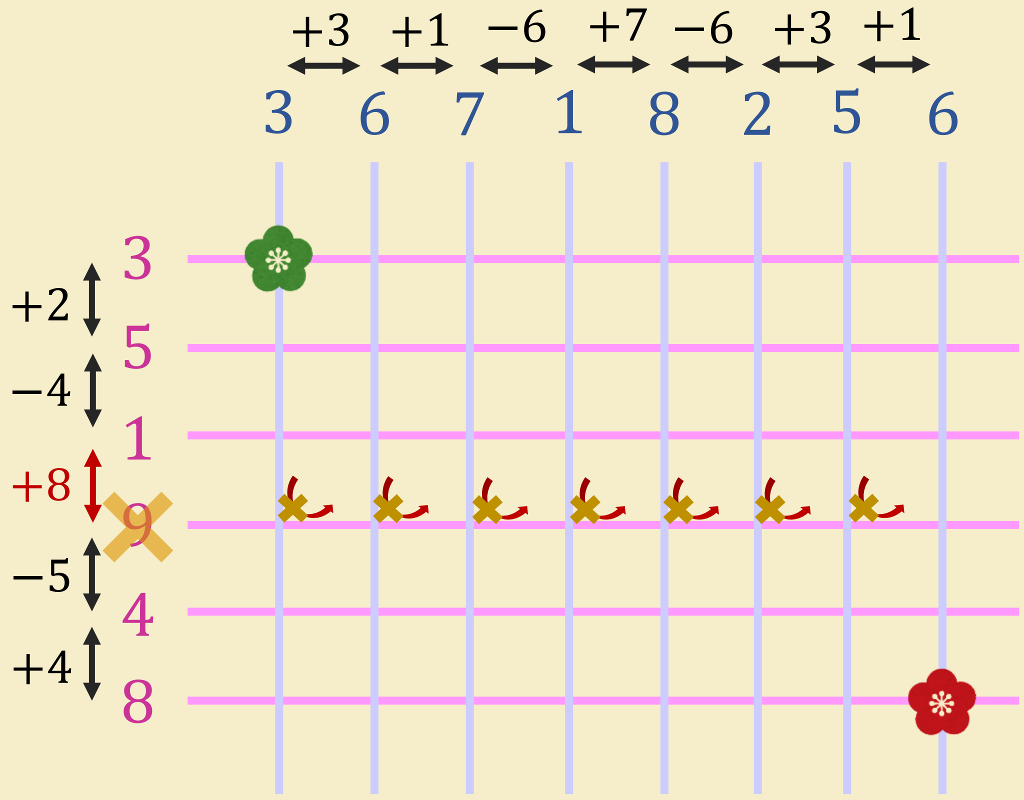
（右図の黒い数は「傾き」を表しています）



小課題 3 — $H, W \leq 100000$

最初からやってみよう！

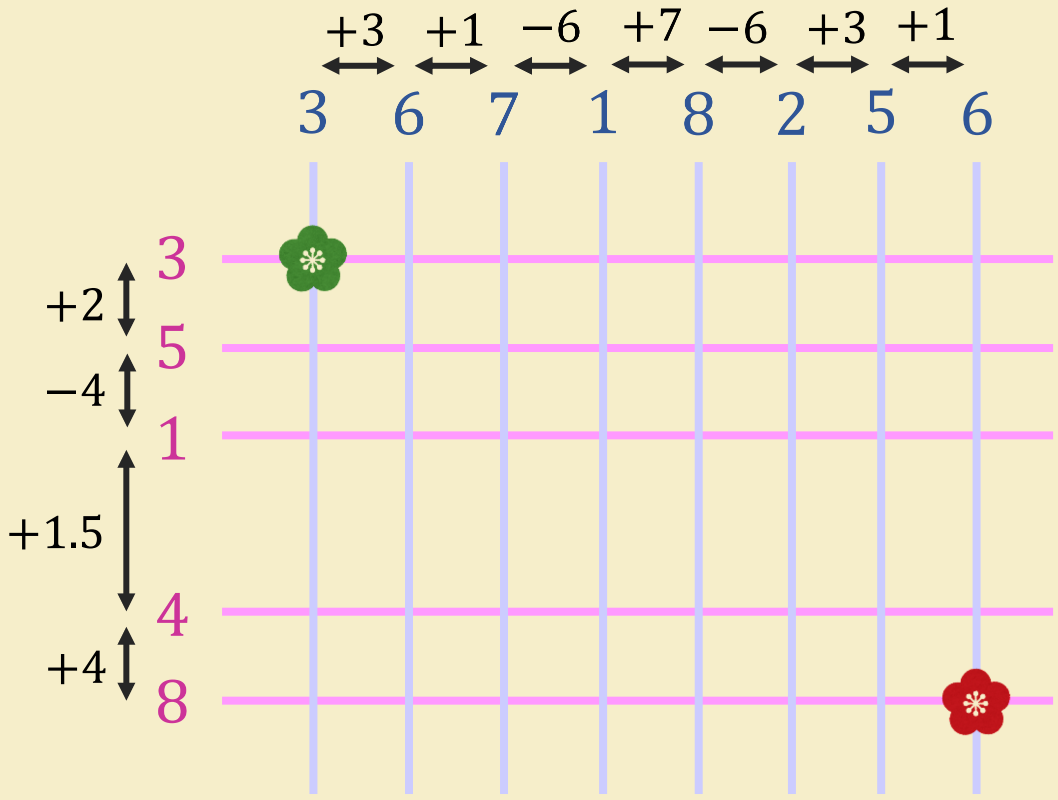
（右図の黒い数は「傾き」を表しています）



小課題 3 — $H, W \leq 100000$

最初からやってみよう！

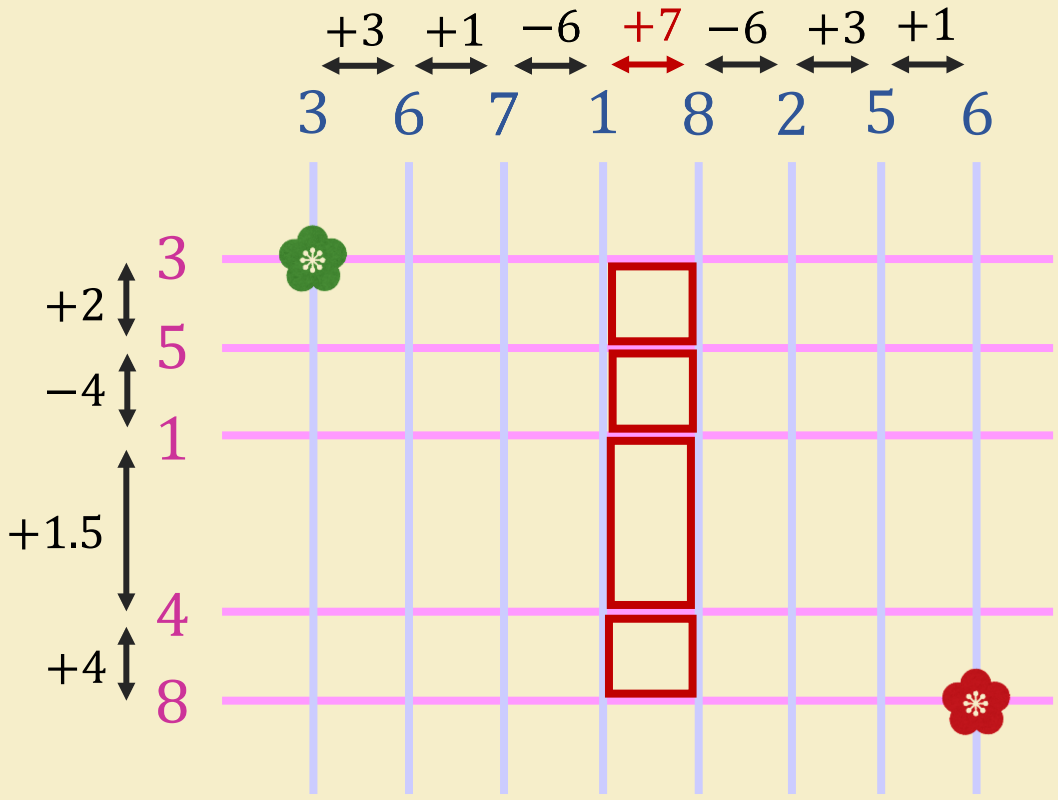
（右図の黒い数は「傾き」を表しています）



小課題 3 — $H, W \leq 100000$

最初からやってみよう!

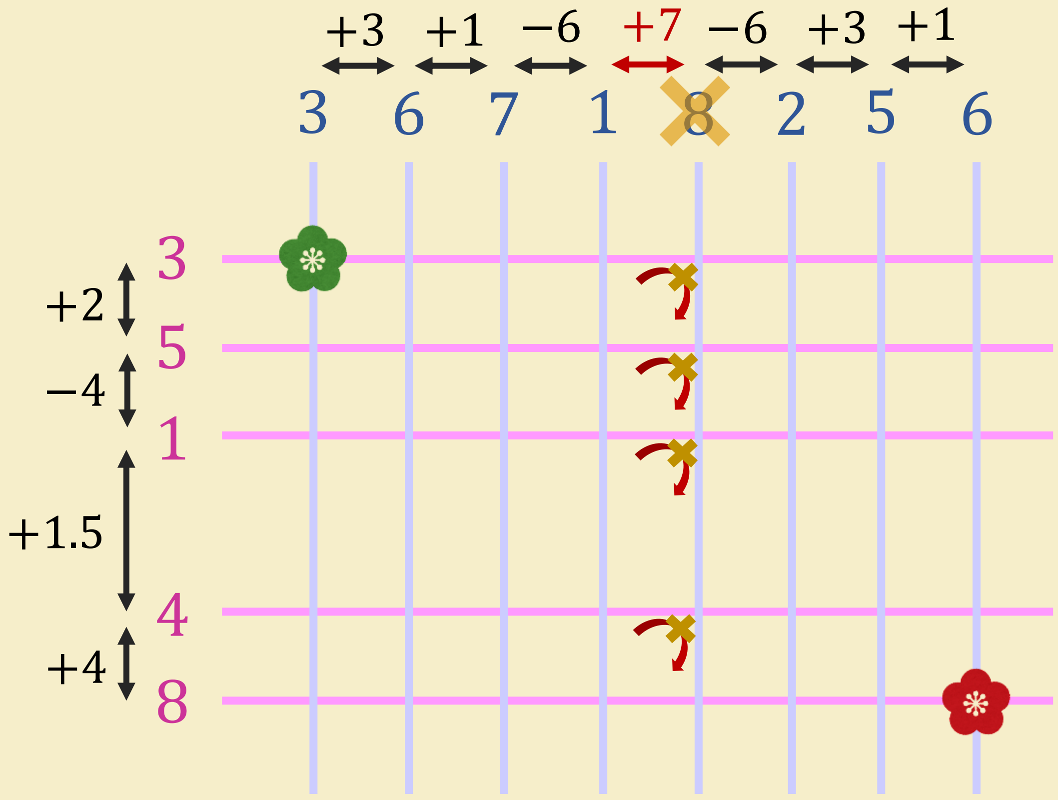
(右図の黒い数は「傾き」を表しています)



小課題 3 — $H, W \leq 100000$

最初からやってみよう!

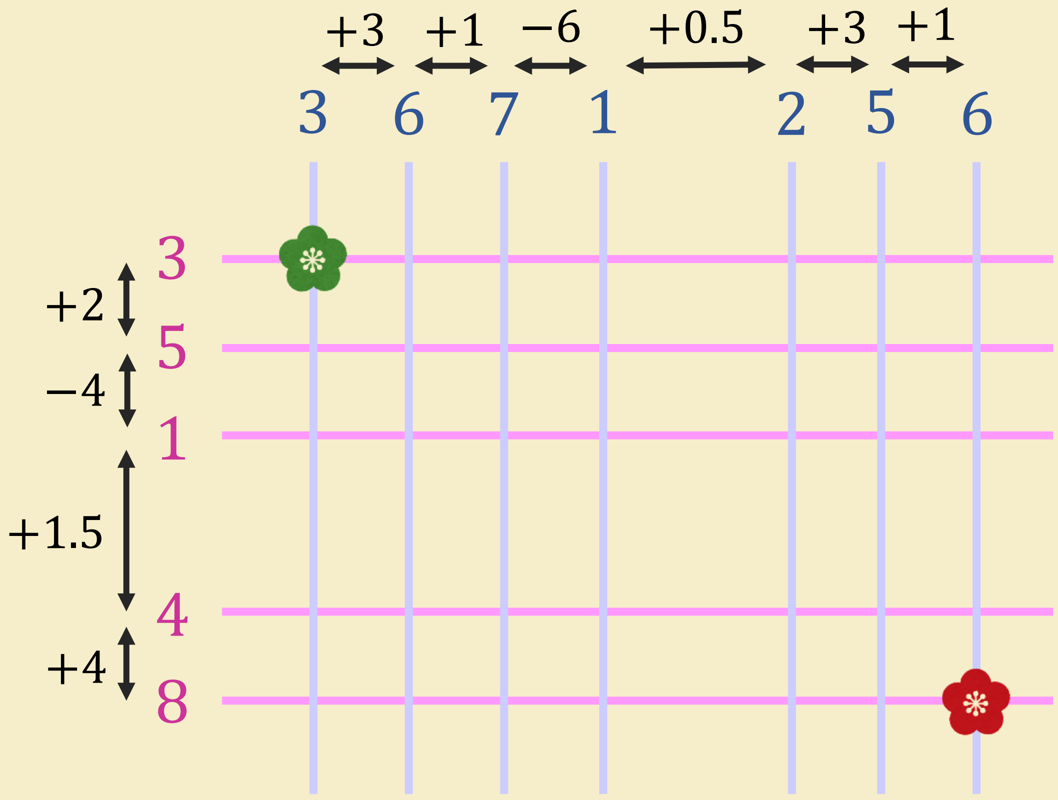
(右図の黒い数は「傾き」を表しています)



小課題 3 — $H, W \leq 100000$

最初からやってみよう！

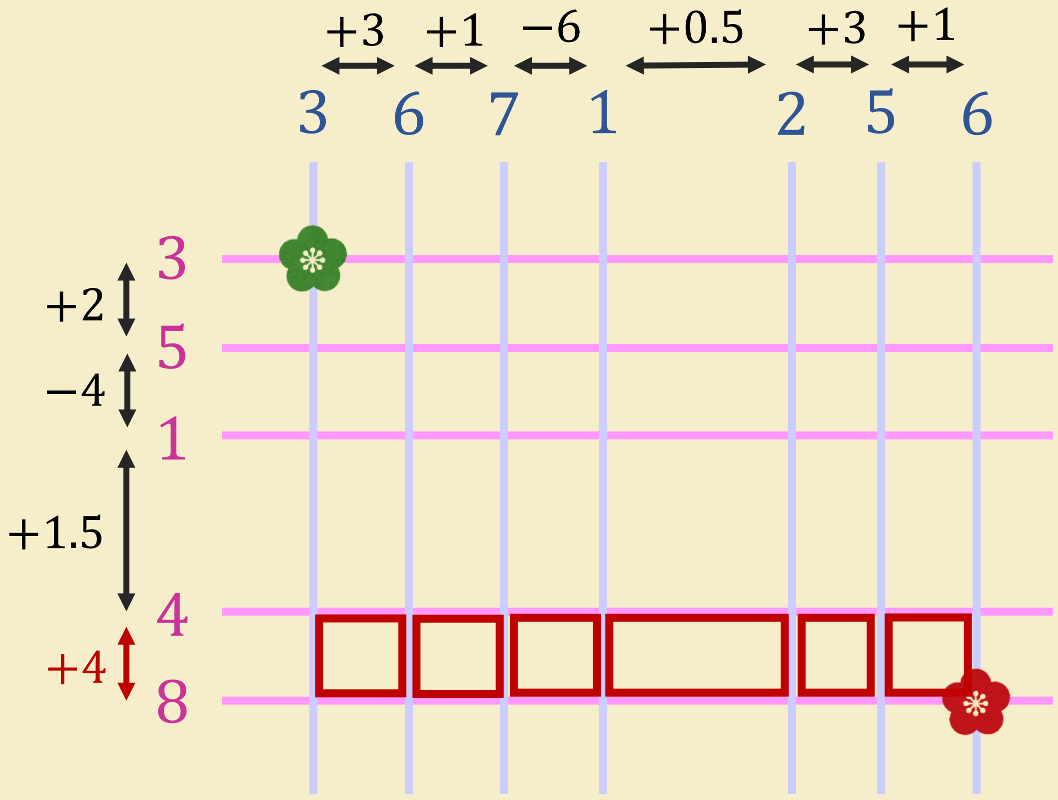
（右図の黒い数は「傾き」を表しています）



小課題 3 — $H, W \leq 100000$

最初からやってみよう！

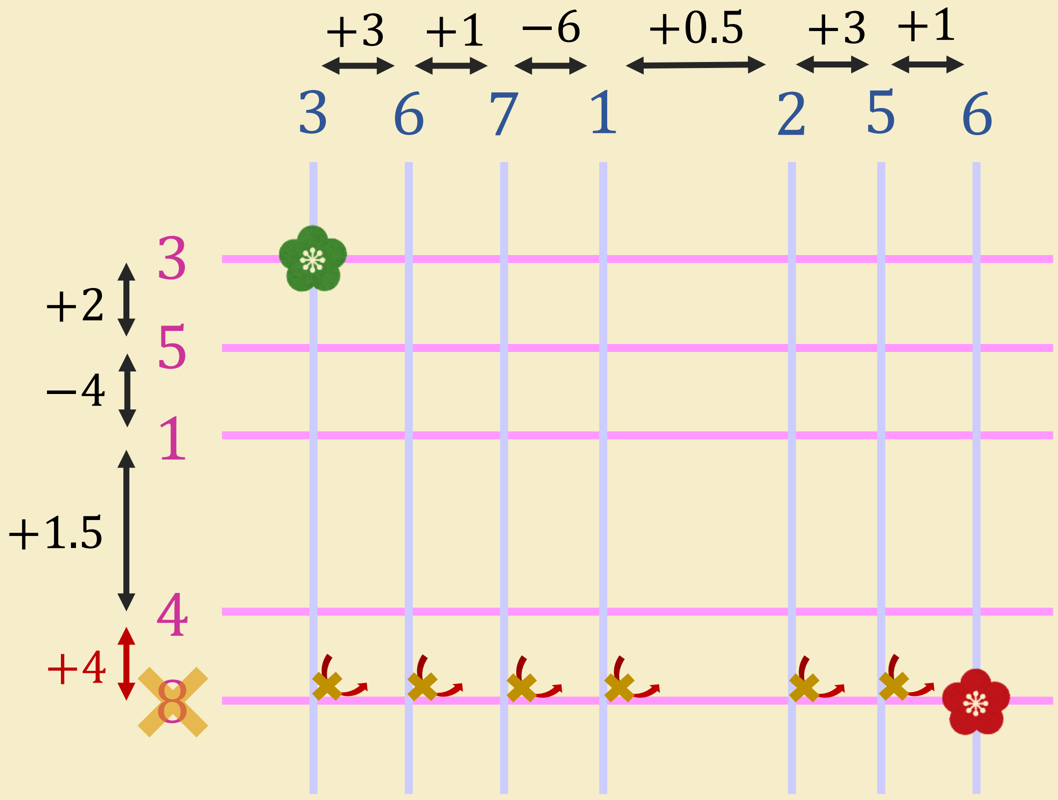
（右図の黒い数は「傾き」を表しています）



小課題 3 — $H, W \leq 100000$

最初からやってみよう！

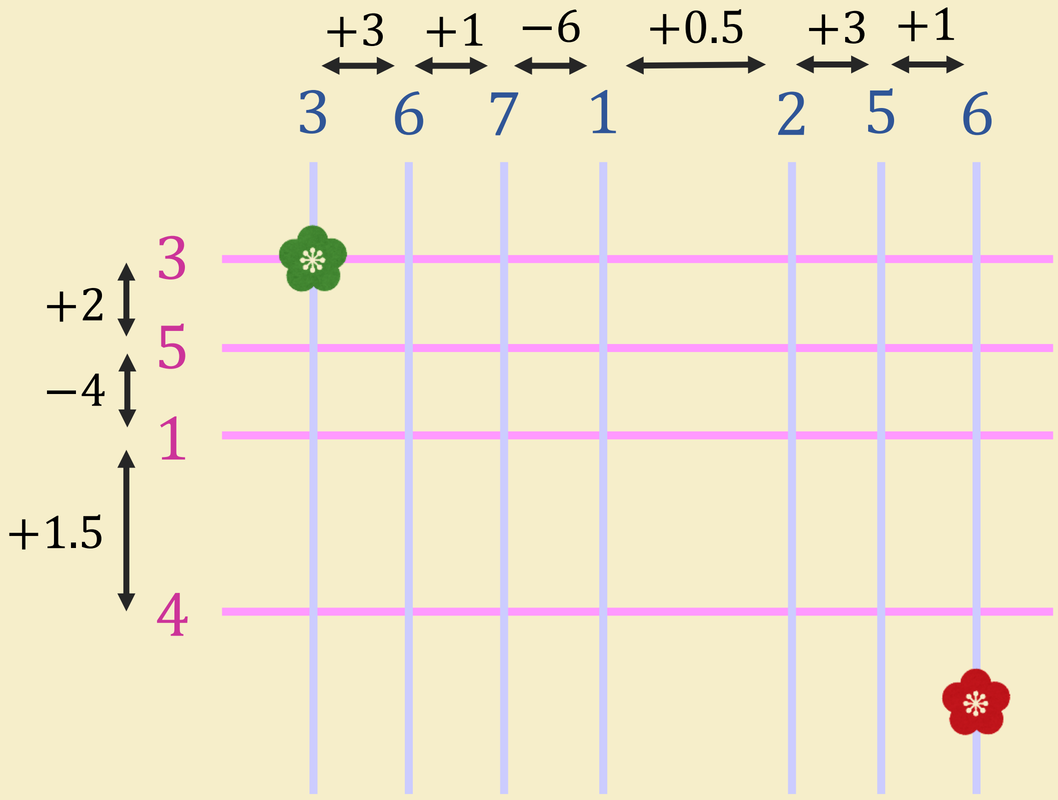
（右図の黒い数は「傾き」を表しています）



小課題 3 — $H, W \leq 100000$

最初からやってみよう！

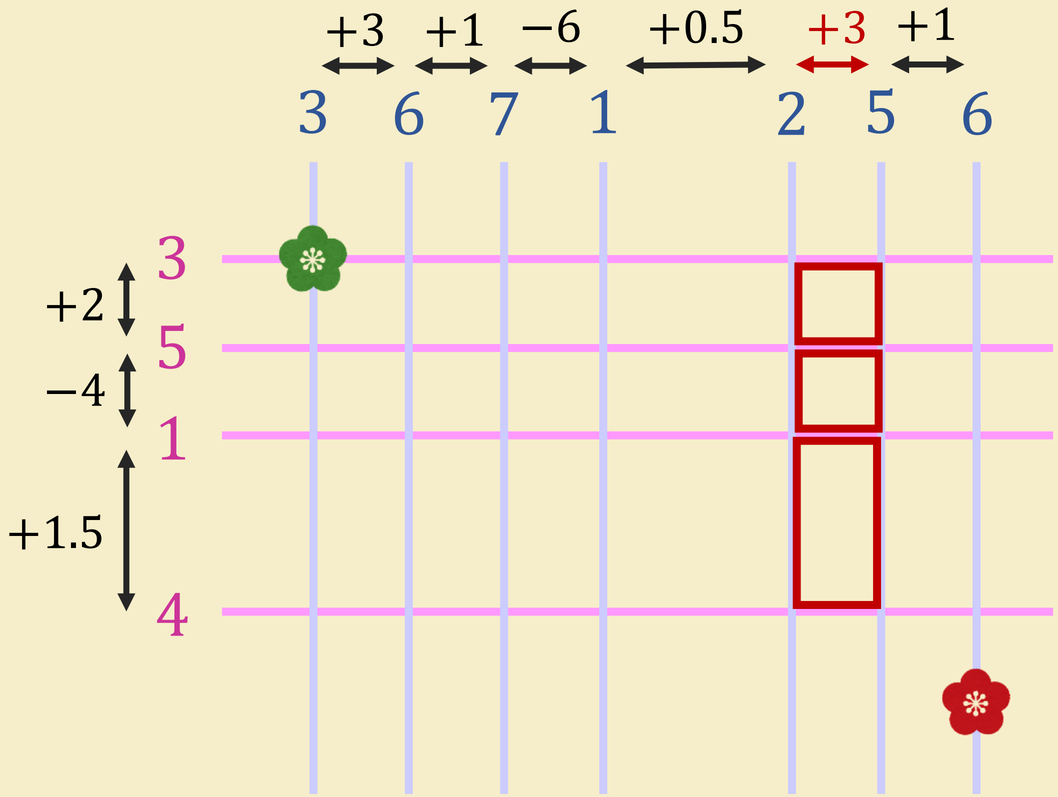
（右図の黒い数は「傾き」を表しています）



小課題 3 — $H, W \leq 100000$

最初からやってみよう！

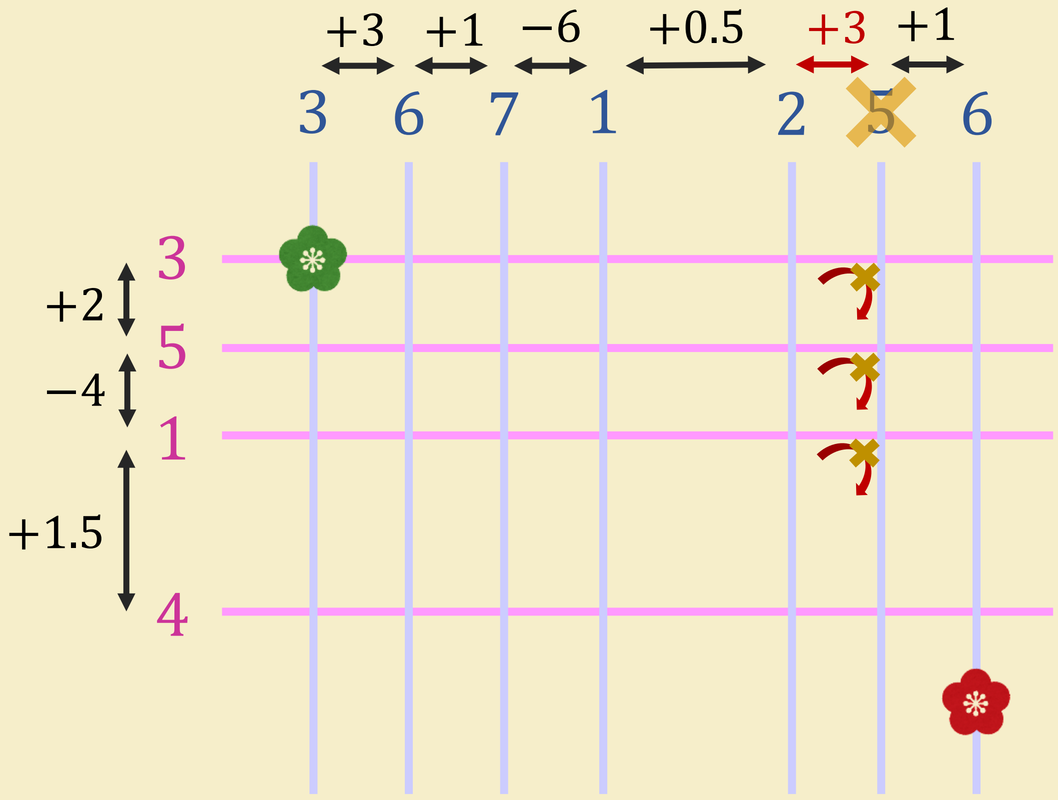
（右図の黒い数は「傾き」を表しています）



小課題 3 — $H, W \leq 100000$

最初からやってみよう！

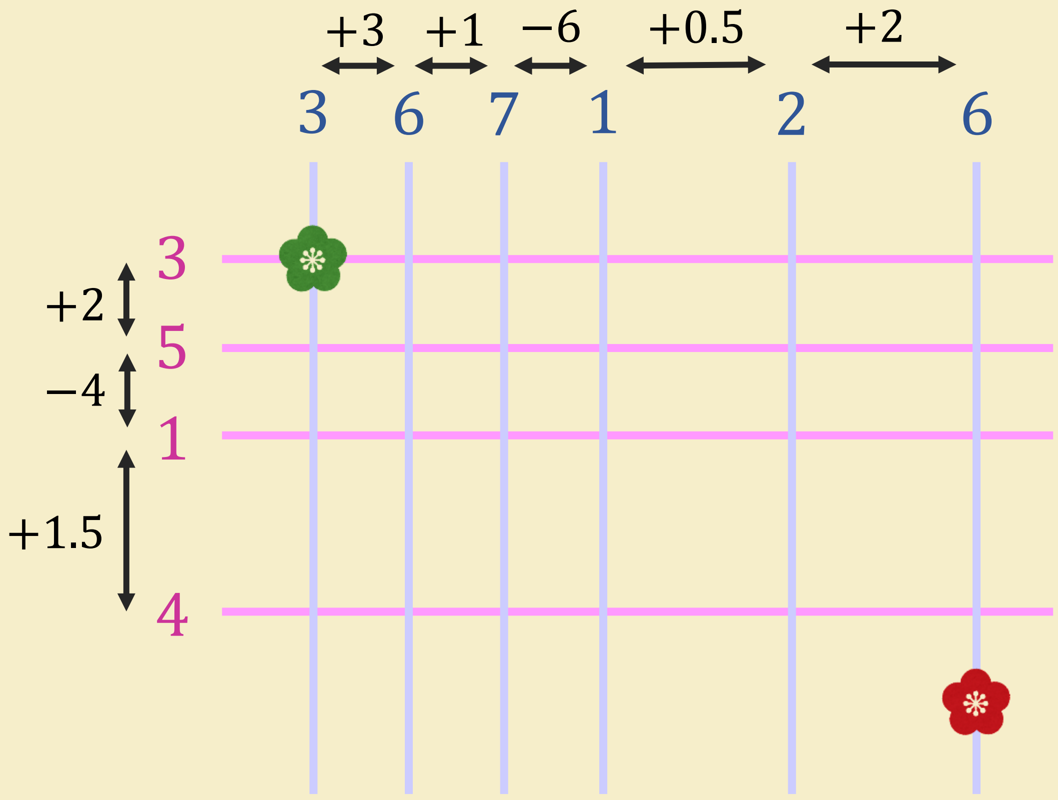
（右図の黒い数は「傾き」を表しています）



小課題 3 — $H, W \leq 100000$

最初からやってみよう！

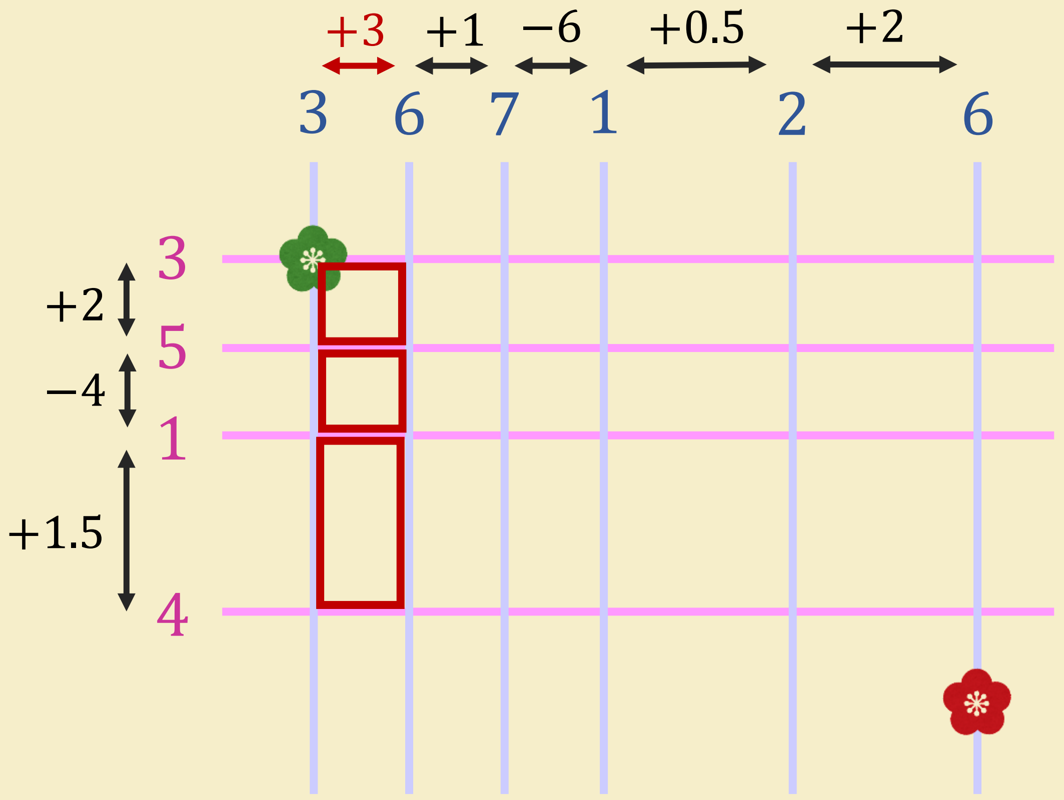
（右図の黒い数は「傾き」を表しています）



小課題 3 — $H, W \leq 100000$

最初からやってみよう！

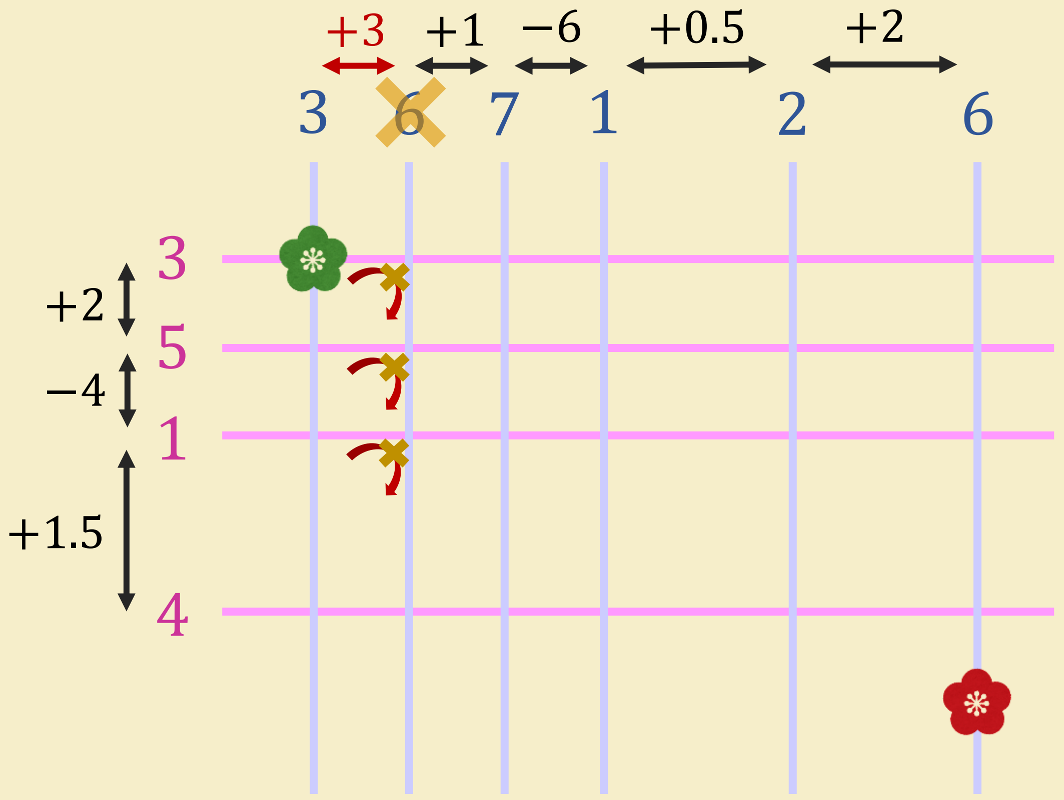
（右図の黒い数は「傾き」を表しています）



小課題 3 — $H, W \leq 100000$

最初からやってみよう！

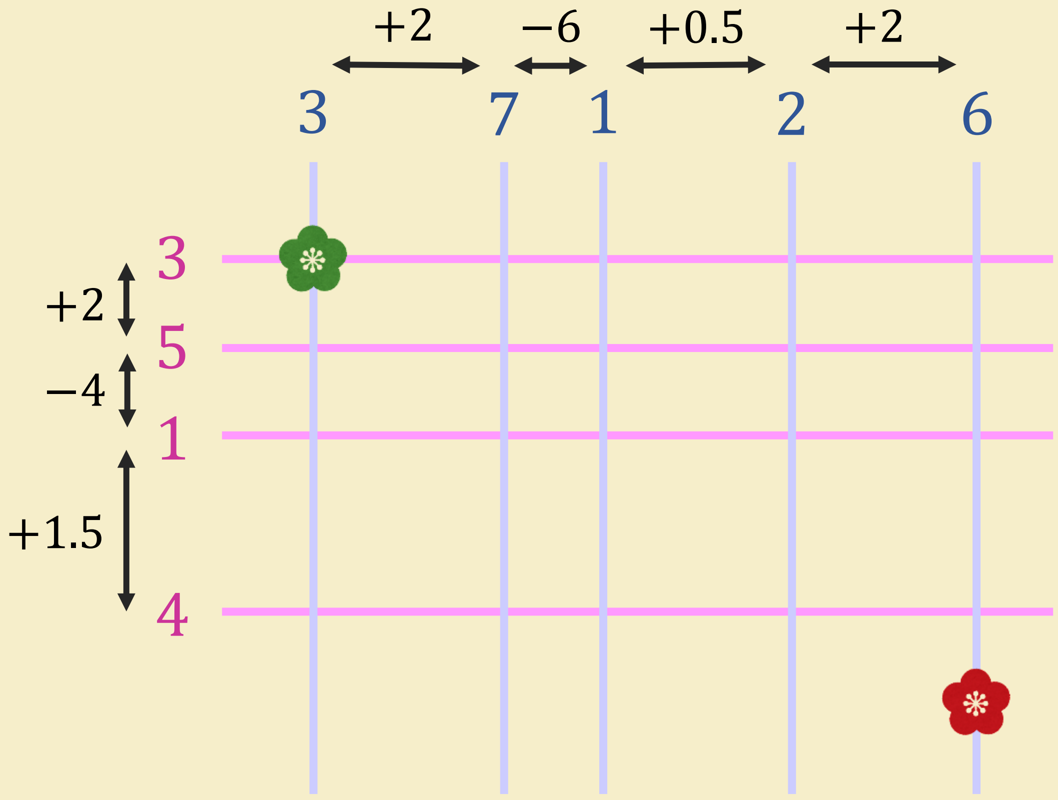
（右図の黒い数は「傾き」を表しています）



小課題 3 — $H, W \leq 100000$

最初からやってみよう！

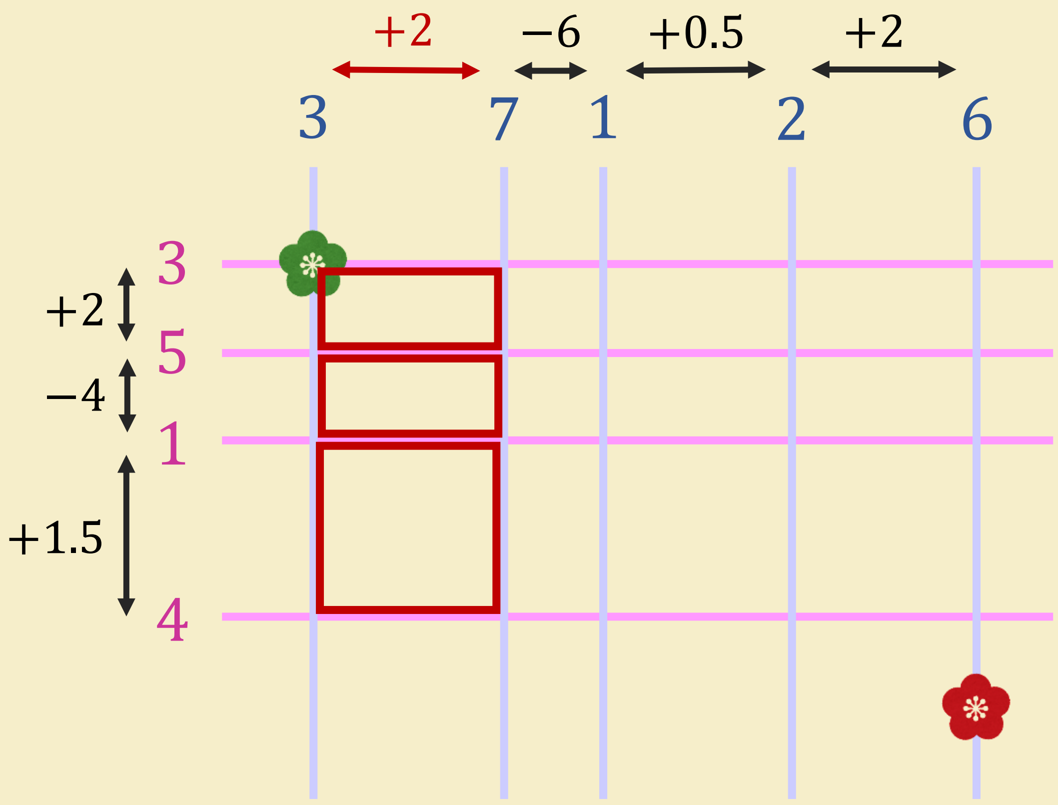
（右図の黒い数は「傾き」を表しています）



小課題 3 — $H, W \leq 100000$

最初からやってみよう！

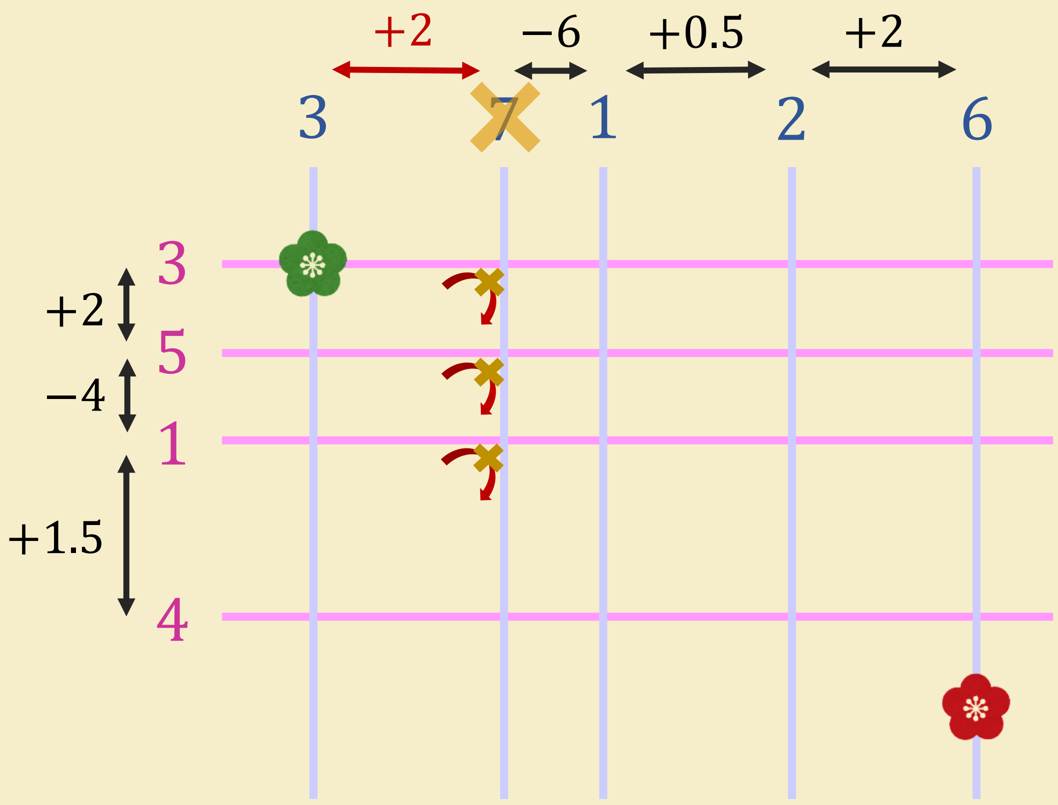
（右図の黒い数は「傾き」を表しています）



小課題 3 — $H, W \leq 100000$

最初からやってみよう！

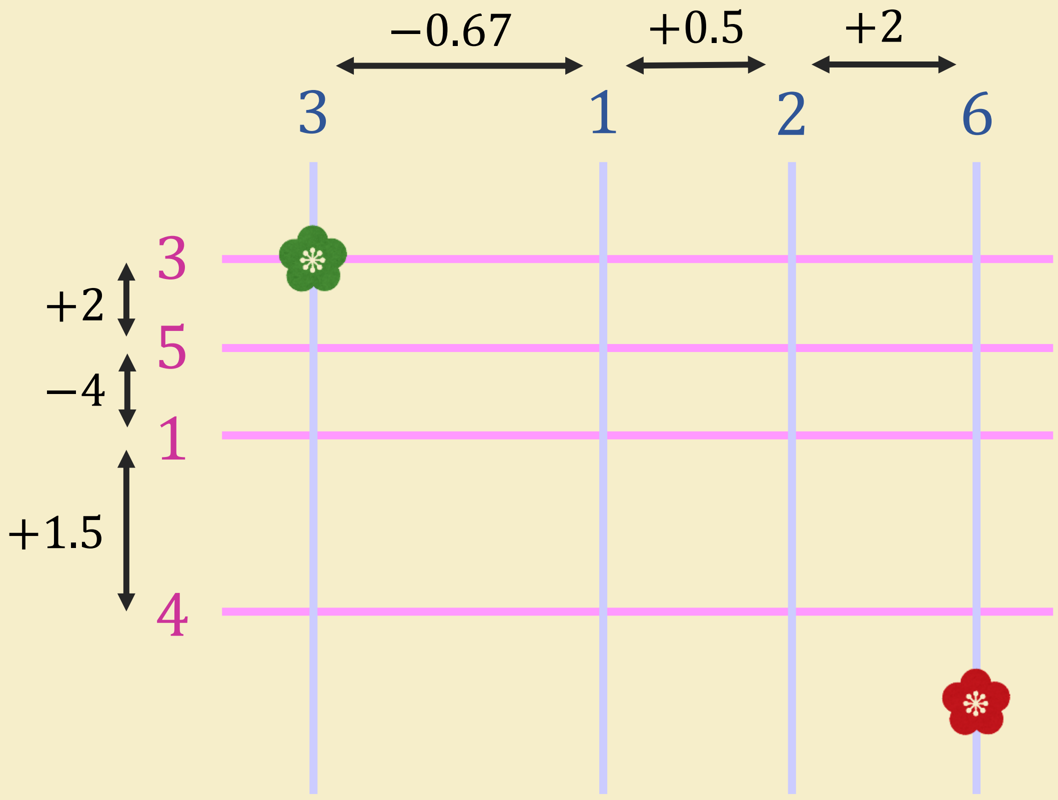
（右図の黒い数は「傾き」を表しています）



小課題 3 — $H, W \leq 100000$

最初からやってみよう！

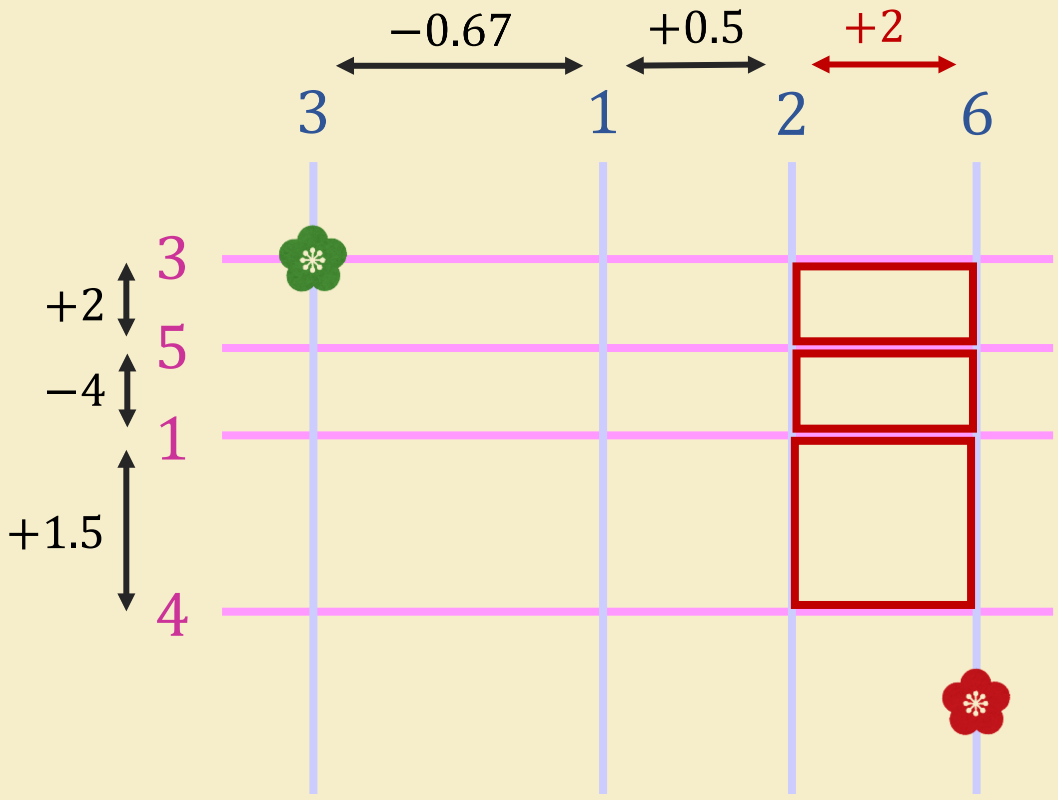
（右図の黒い数は「傾き」を表しています）



小課題 3 — $H, W \leq 100000$

最初からやってみよう！

（右図の黒い数は「傾き」を表しています）

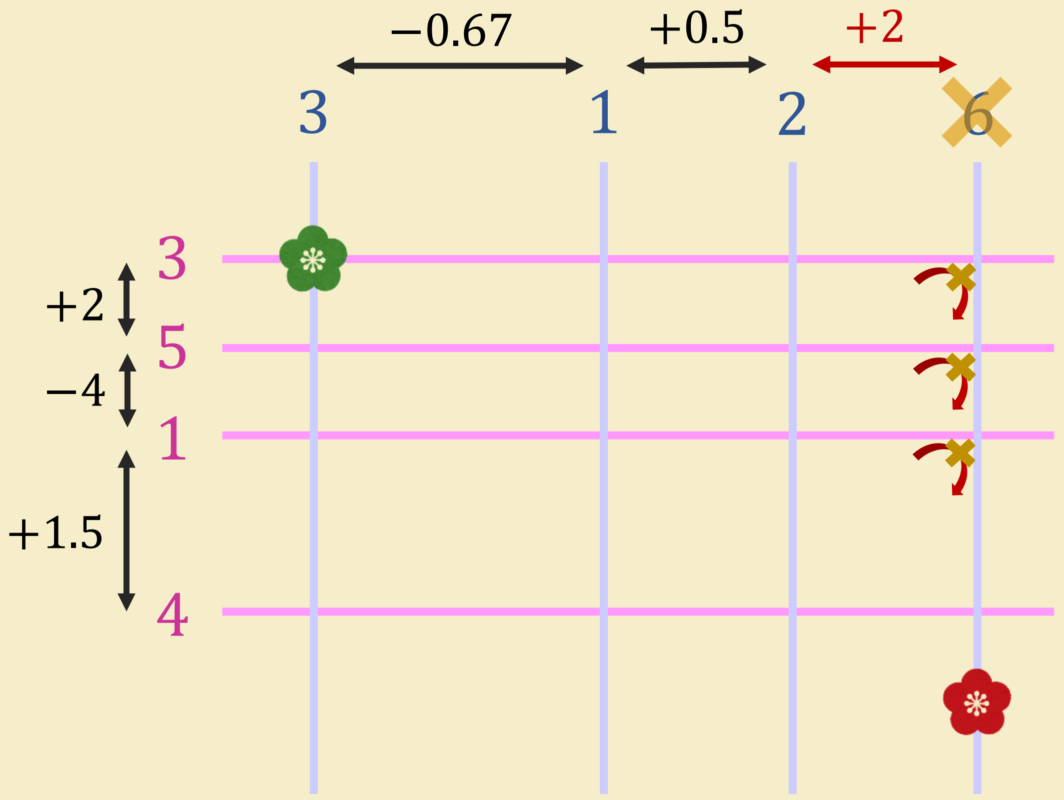


小課題 3 — $H, W \leq 100000$

最初からやってみよう！

（右図の黒い数は「傾き」を表しています）

四角に対して条件が適用されるので
道路の最後の部分だけ残ることに注意

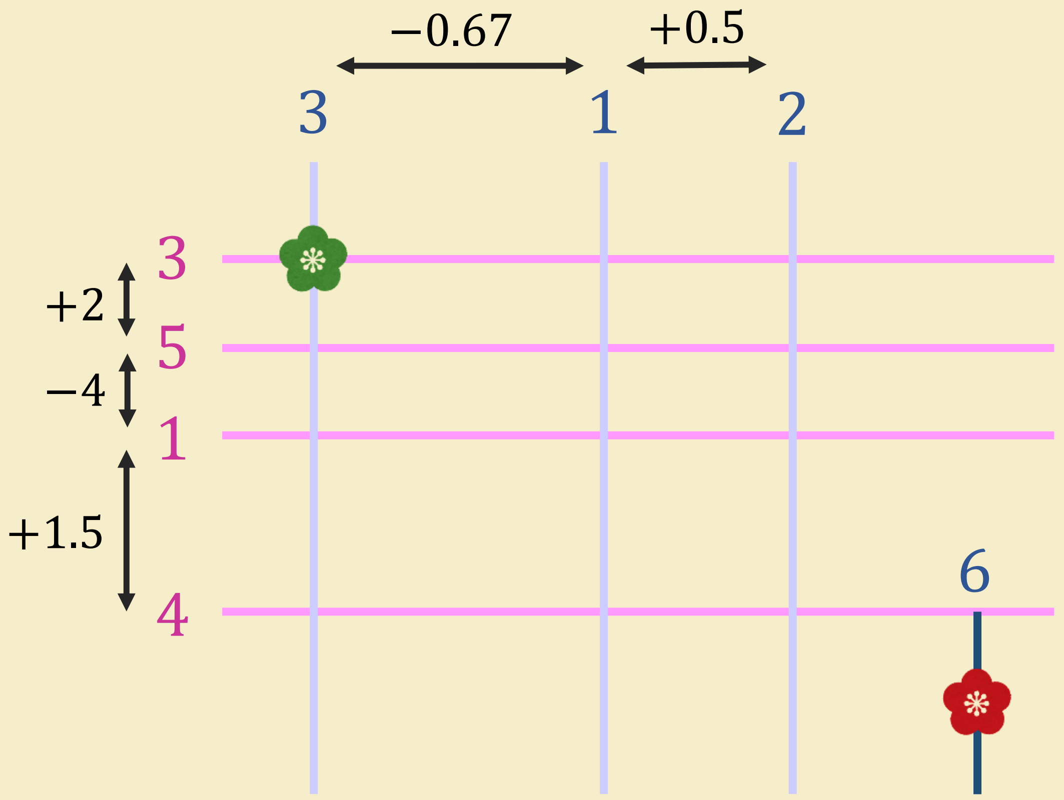


小課題 3 — $H, W \leq 100000$

最初からやってみよう！

（右図の黒い数は「傾き」を表しています）

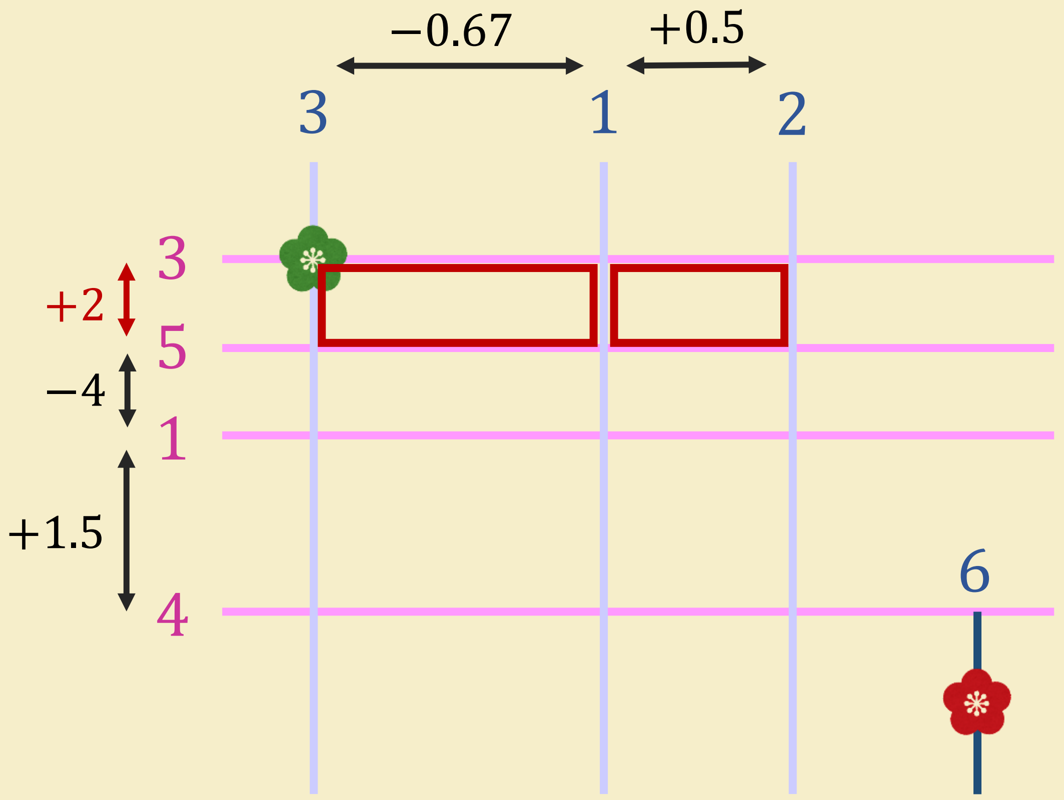
四角に対して条件が適用されるので
道路の最後の部分だけ残ることに注意



小課題 3 — $H, W \leq 100000$

最初からやってみよう！

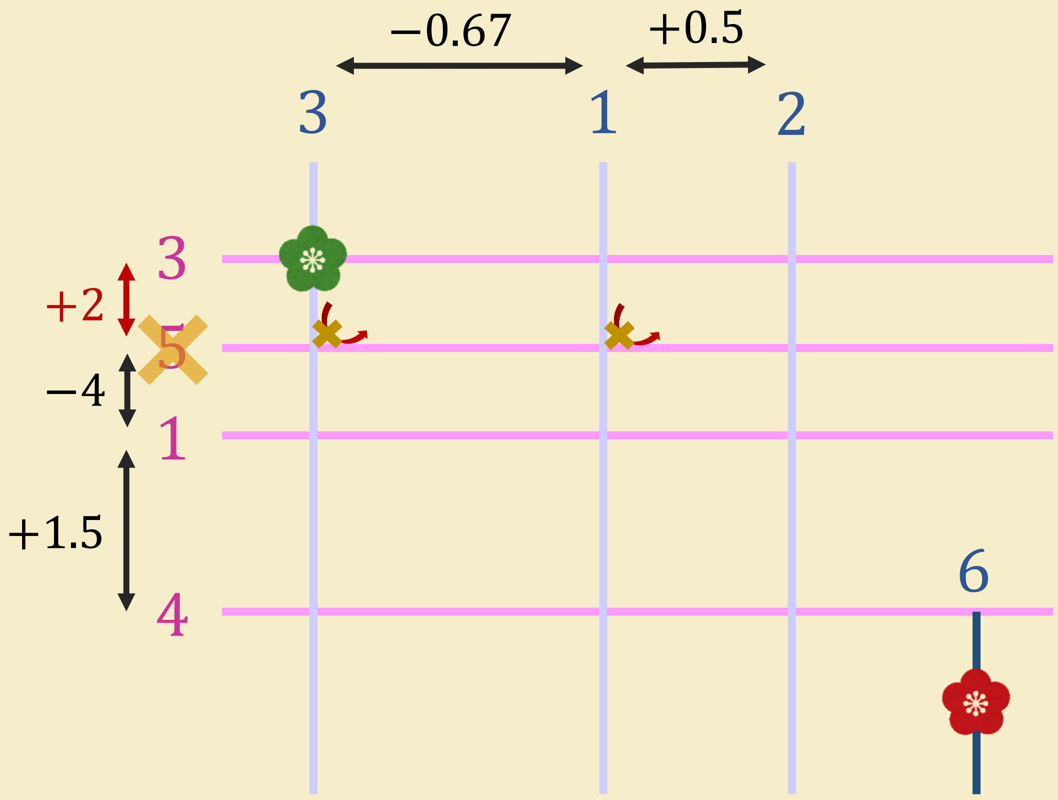
（右図の黒い数は「傾き」を表しています）



小課題 3 — $H, W \leq 100000$

最初からやってみよう!

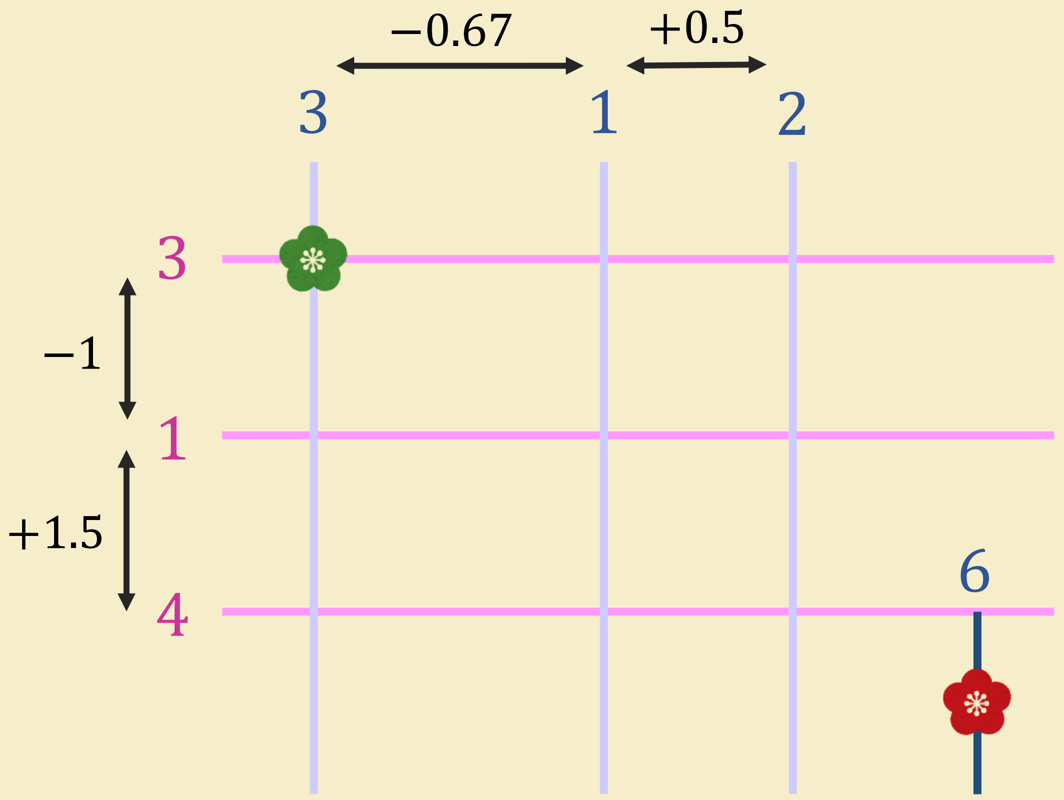
（右図の黒い数は「傾き」を表しています）



小課題 3 — $H, W \leq 100000$

最初からやってみよう！

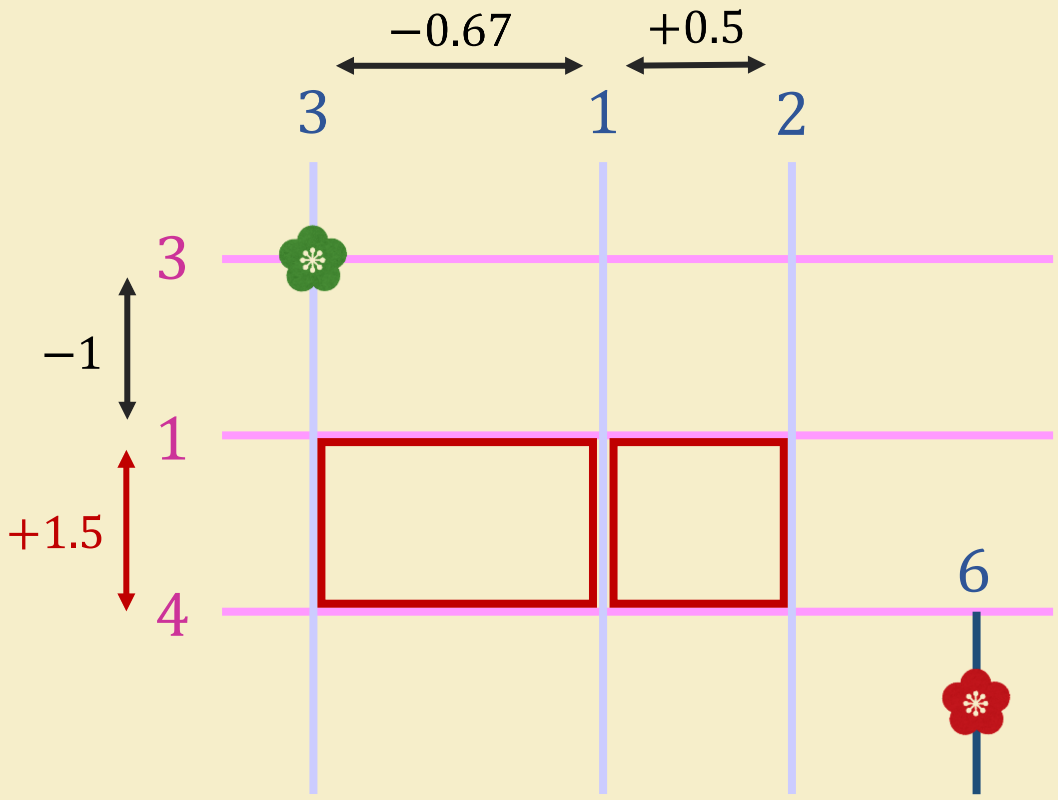
（右図の黒い数は「傾き」を表しています）



小課題 3 — $H, W \leq 100000$

最初からやってみよう！

（右図の黒い数は「傾き」を表しています）

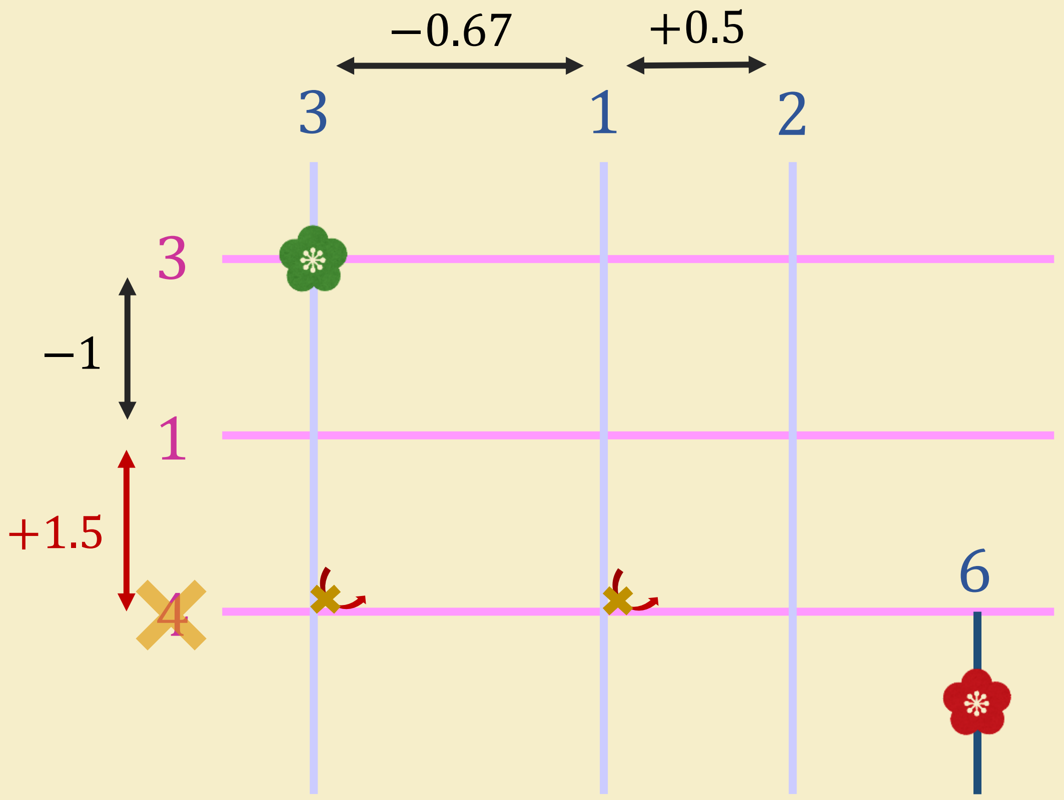


小課題 3 — $H, W \leq 100000$

最初からやってみよう！

（右図の黒い数は「傾き」を表しています）

四角に対して条件が適用されるので
道路の最後の部分だけ残ることに注意

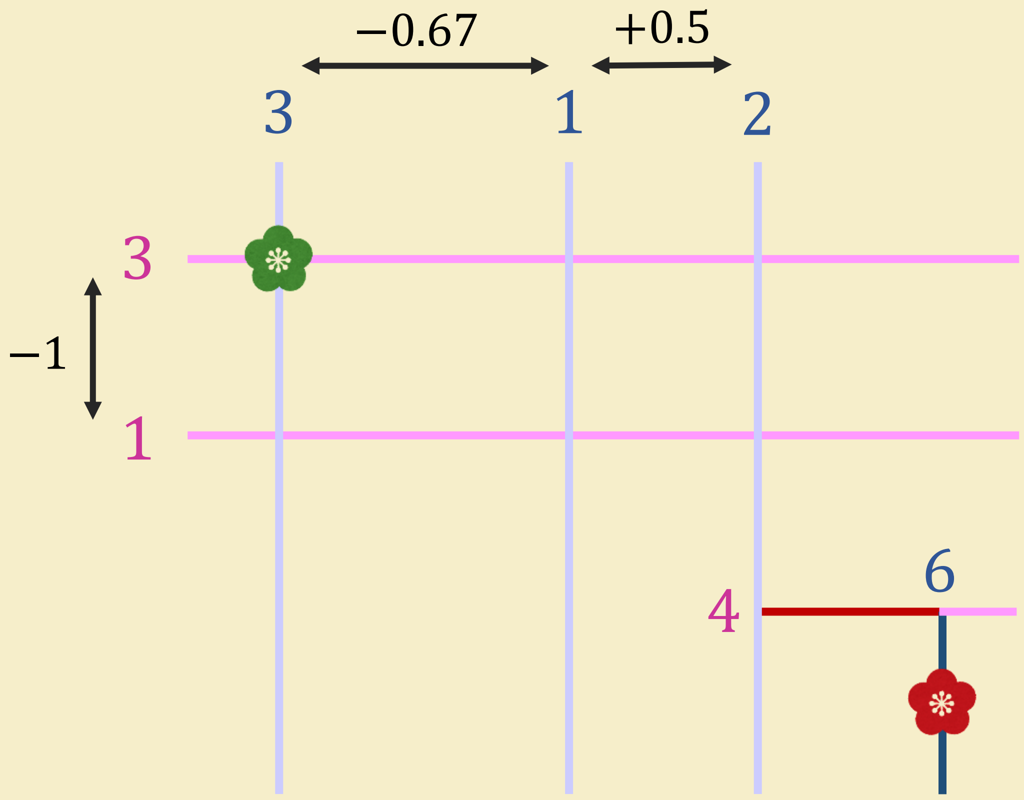


小課題 3 — $H, W \leq 100000$

最初からやってみよう！

（右図の黒い数は「傾き」を表しています）

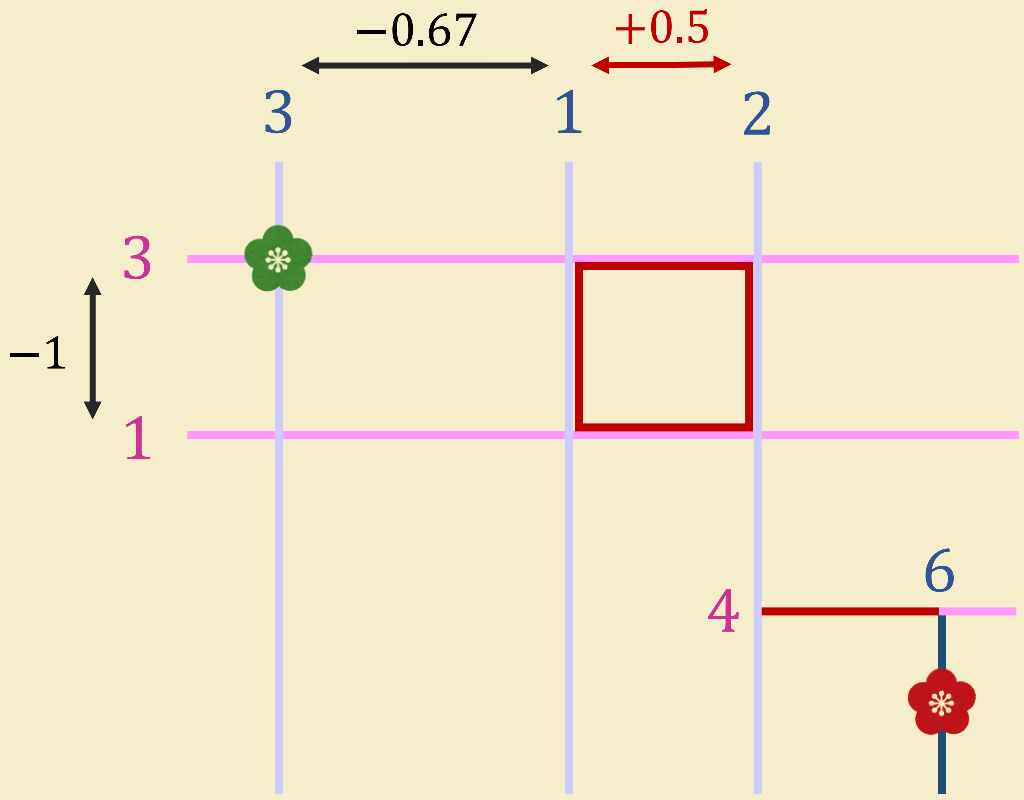
四角に対して条件が適用されるので
道路の最後の部分だけ残ることに注意



小課題 3 — $H, W \leq 100000$

最初からやってみよう！

（右図の黒い数は「傾き」を表しています）

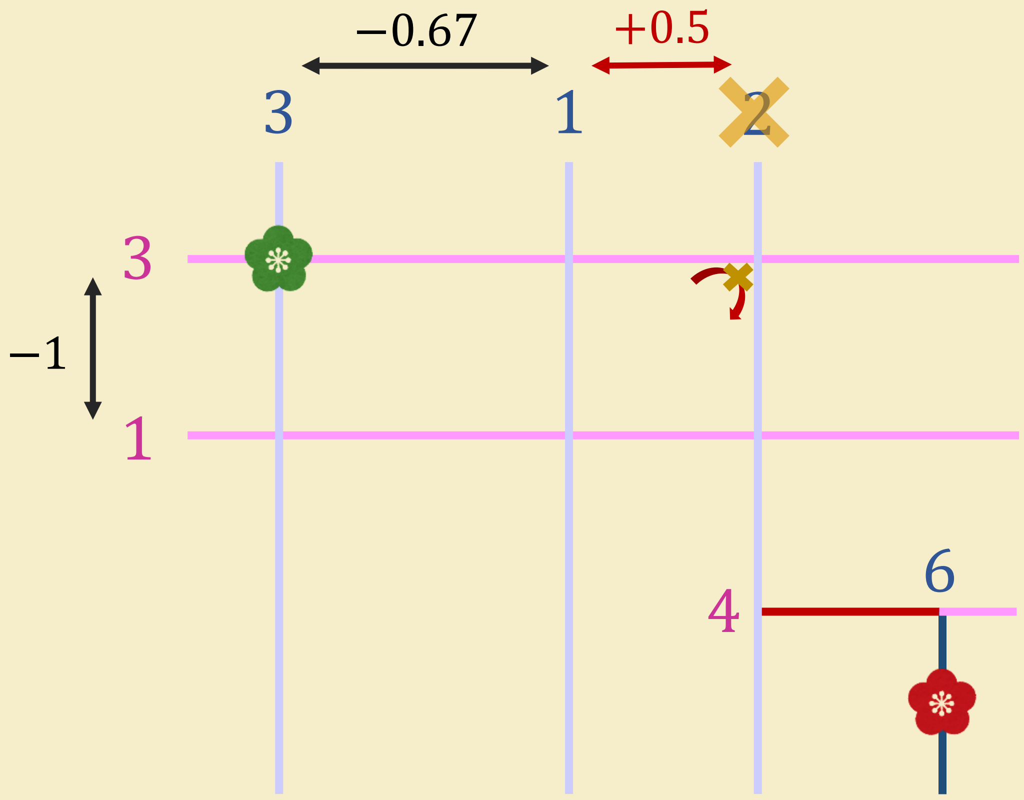


小課題 3 — $H, W \leq 100000$

最初からやってみよう！

（右図の黒い数は「傾き」を表しています）

四角に対して条件が適用されるので
道路の最後の部分だけ残ることに注意

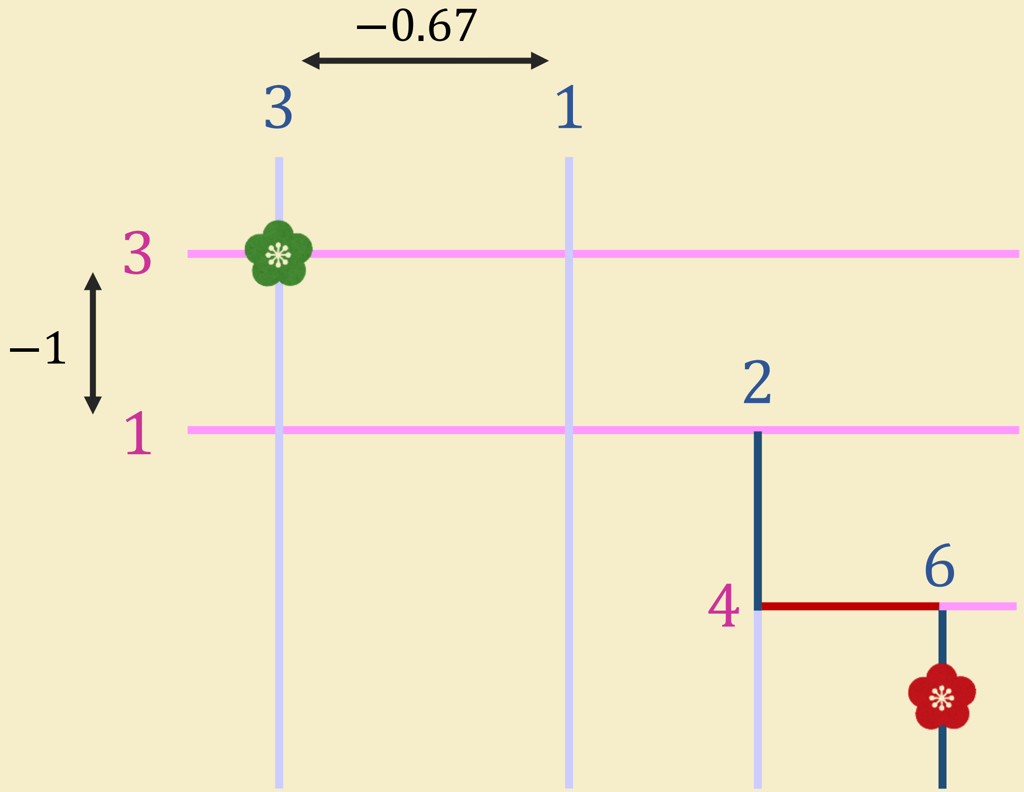


小課題 3 — $H, W \leq 100000$

最初からやってみよう！

（右図の黒い数は「傾き」を表しています）

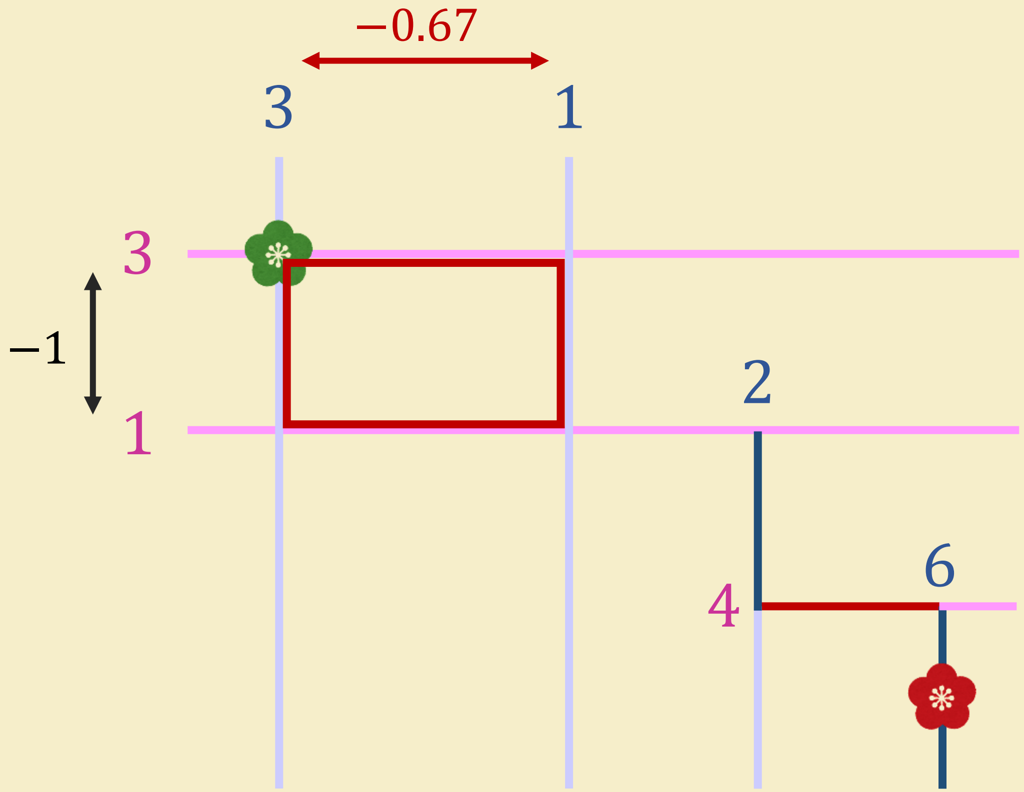
四角に対して条件が適用されるので
道路の最後の部分だけ残ることに注意



小課題 3 — $H, W \leq 100000$

最初からやってみよう！

（右図の黒い数は「傾き」を表しています）

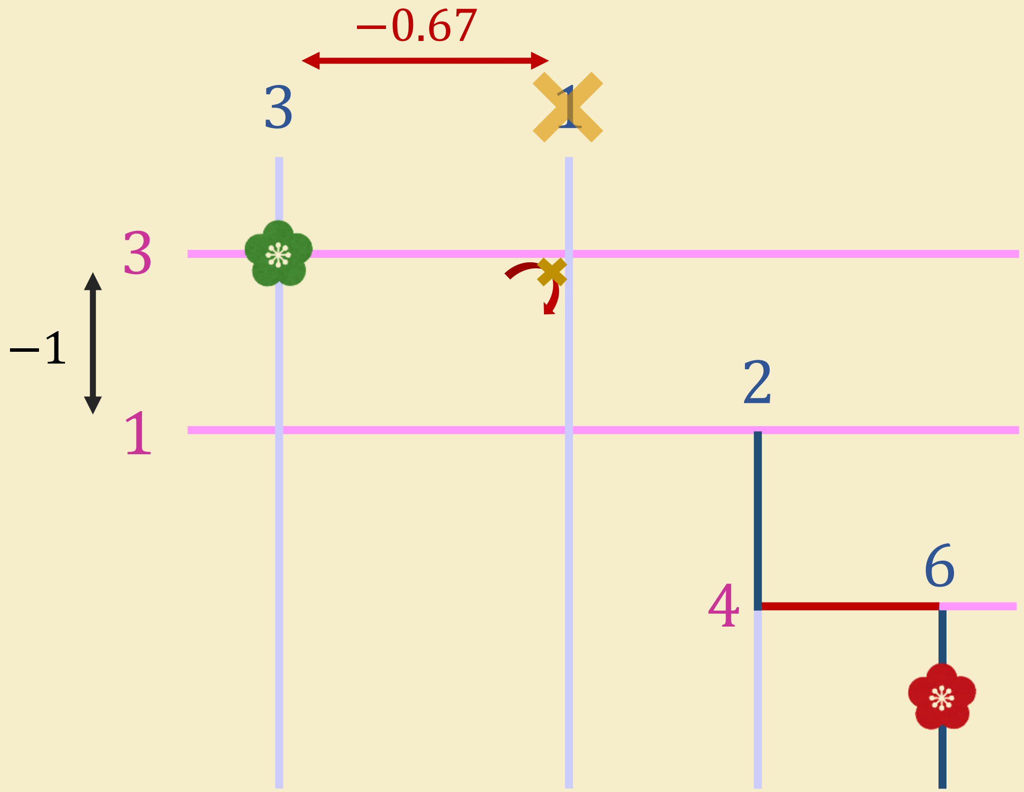


小課題 3 — $H, W \leq 100000$

最初からやってみよう！

（右図の黒い数は「傾き」を表しています）

四角に対して条件が適用されるので
道路の最後の部分だけ残ることに注意

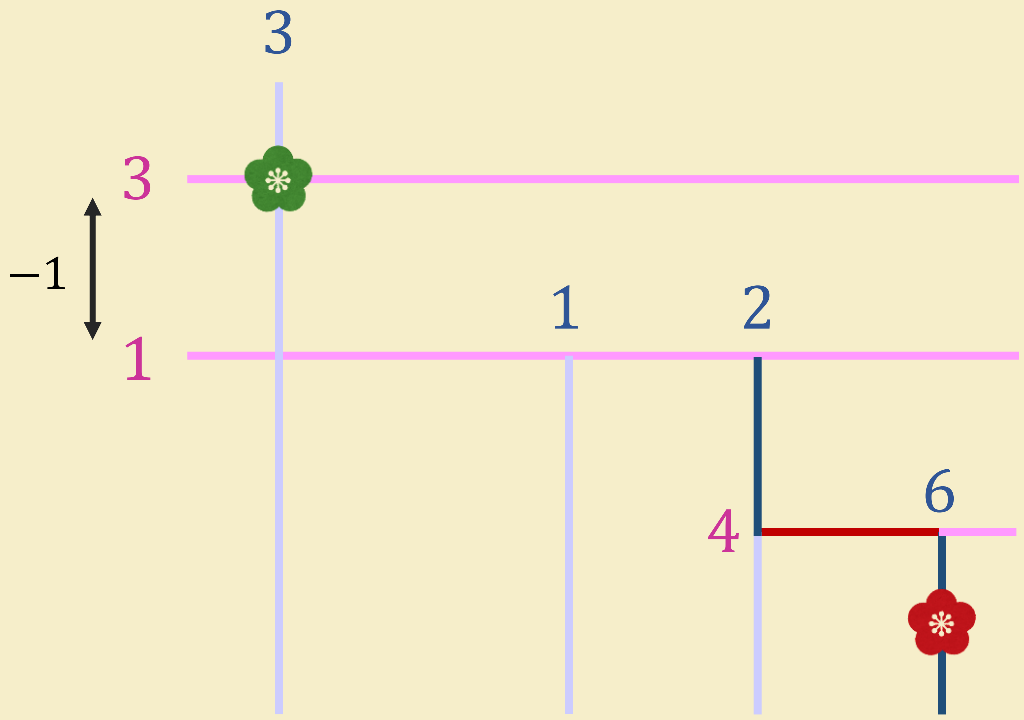


小課題 3 — $H, W \leq 100000$

最初からやってみよう！

（右図の黒い数は「傾き」を表しています）

四角に対して条件が適用されるので
道路の最後の部分だけ残ることに注意

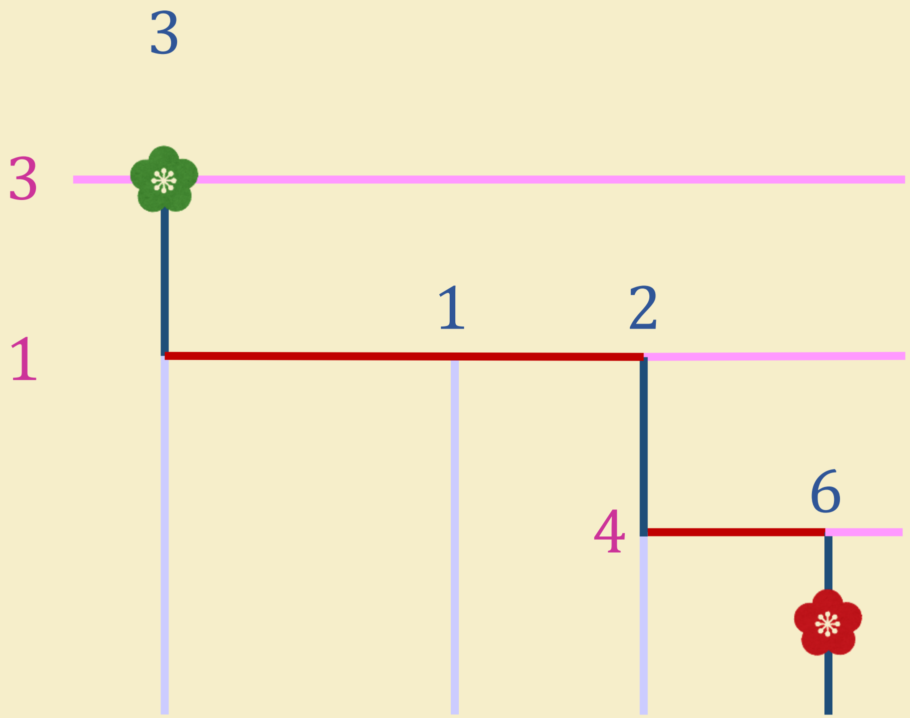


小課題 3 — $H, W \leq 100000$

最初からやってみよう！

（右図の黒い数は「傾き」を表しています）

→ これで経路が決まった！！！



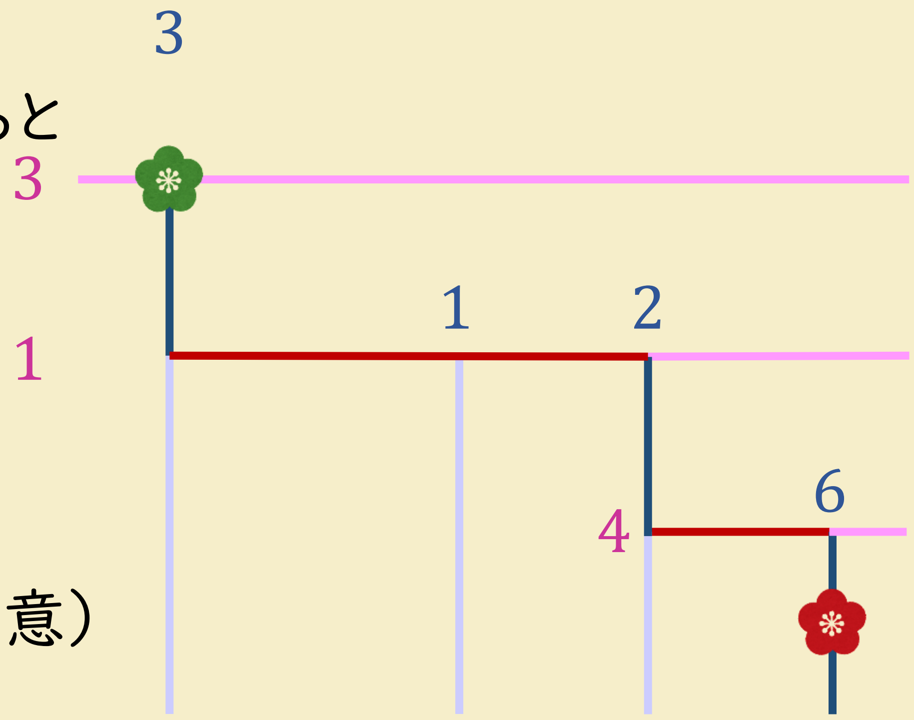
小課題 3 — $H, W \leq 100000$

解法のまとめ

このように、傾きが最小の行／列に条件を適用し続けると
最終的には必ず 1 つの経路ができる

このシミュレーションは、std::set などを用いて
全体計算量 $O\big((H+W)\log(H+W)\big)$ でできる

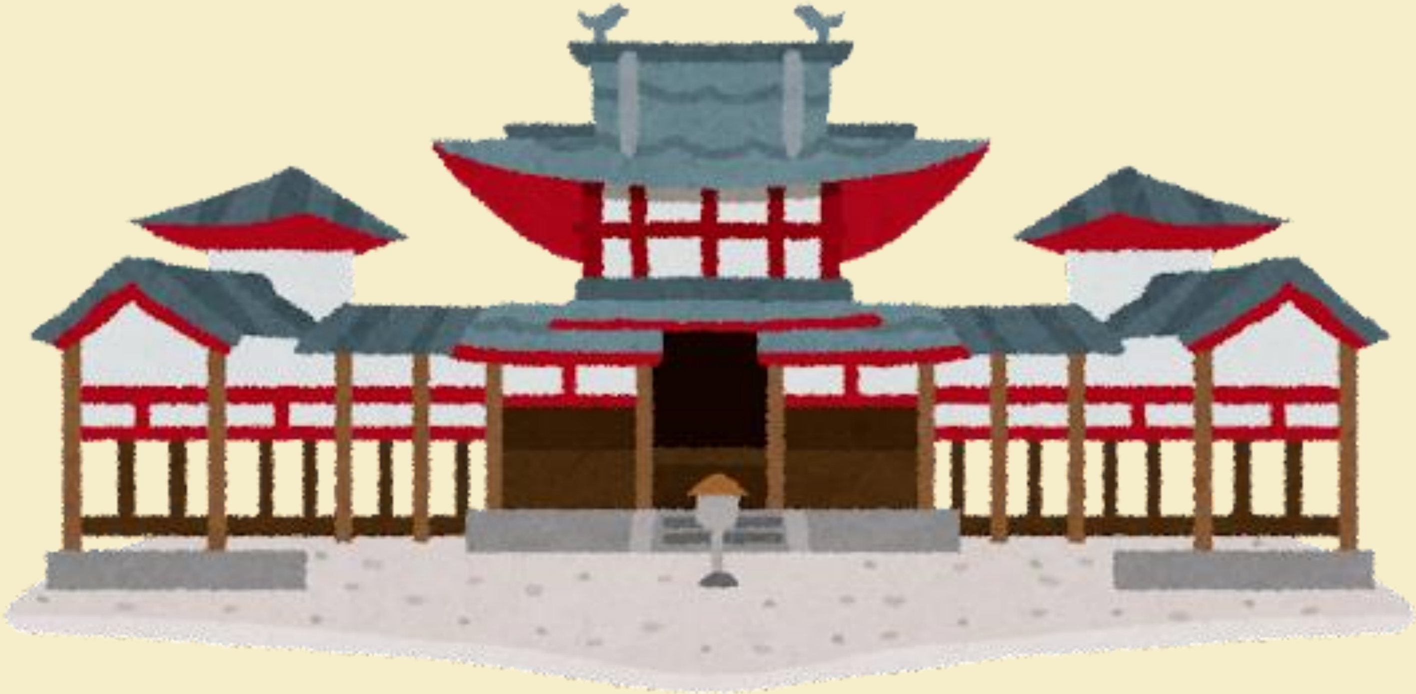
（一番右・下の値が消されたときに経路が残ることに注意）



小課題 3 — $H, W \leq 100000$

解法のまとめ

よって、この問題は $O\big((H+W)\log(H+W)\big)$ で解けた！

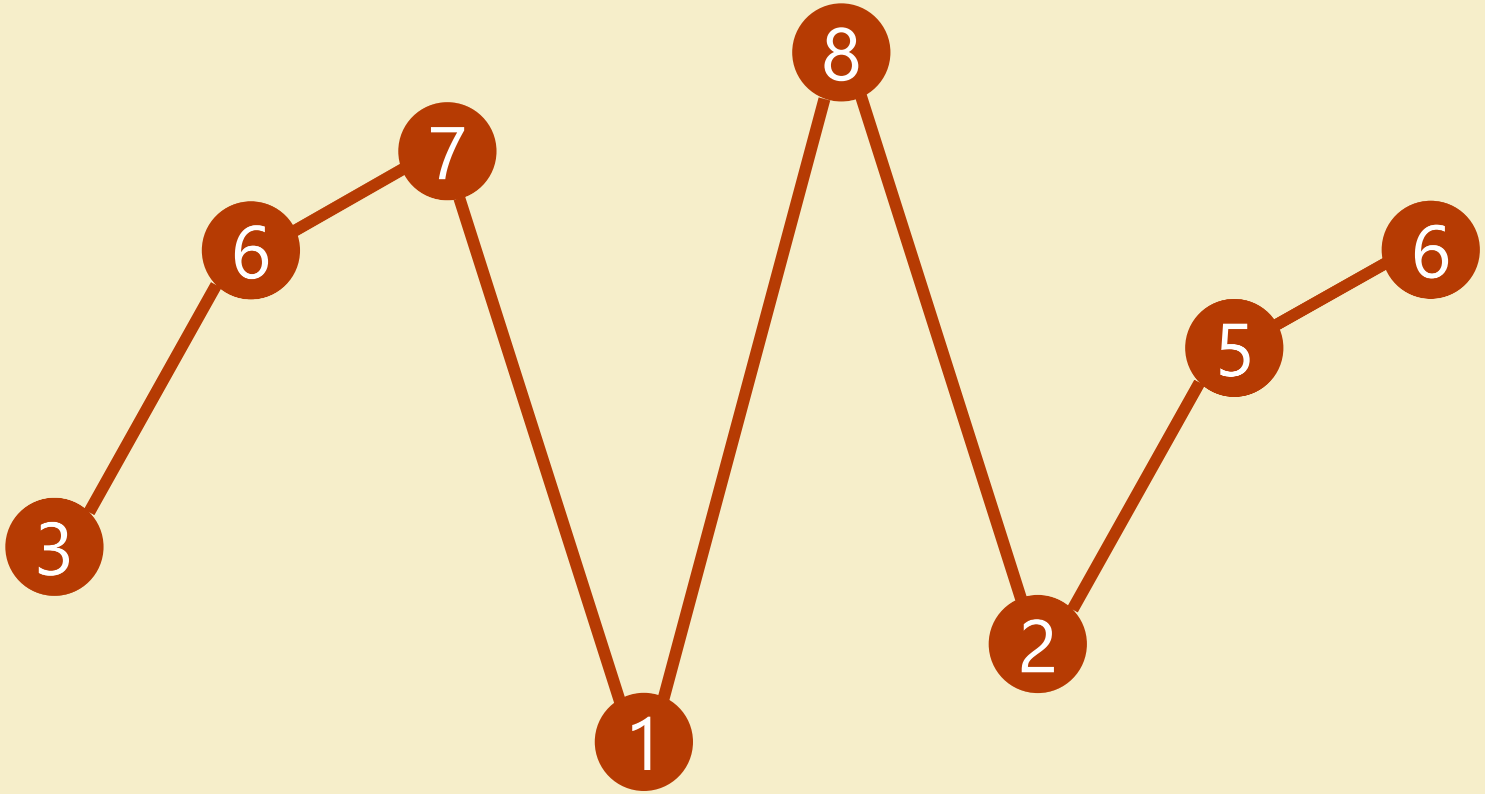


別解

計算量 $O(H+W)$ 解法

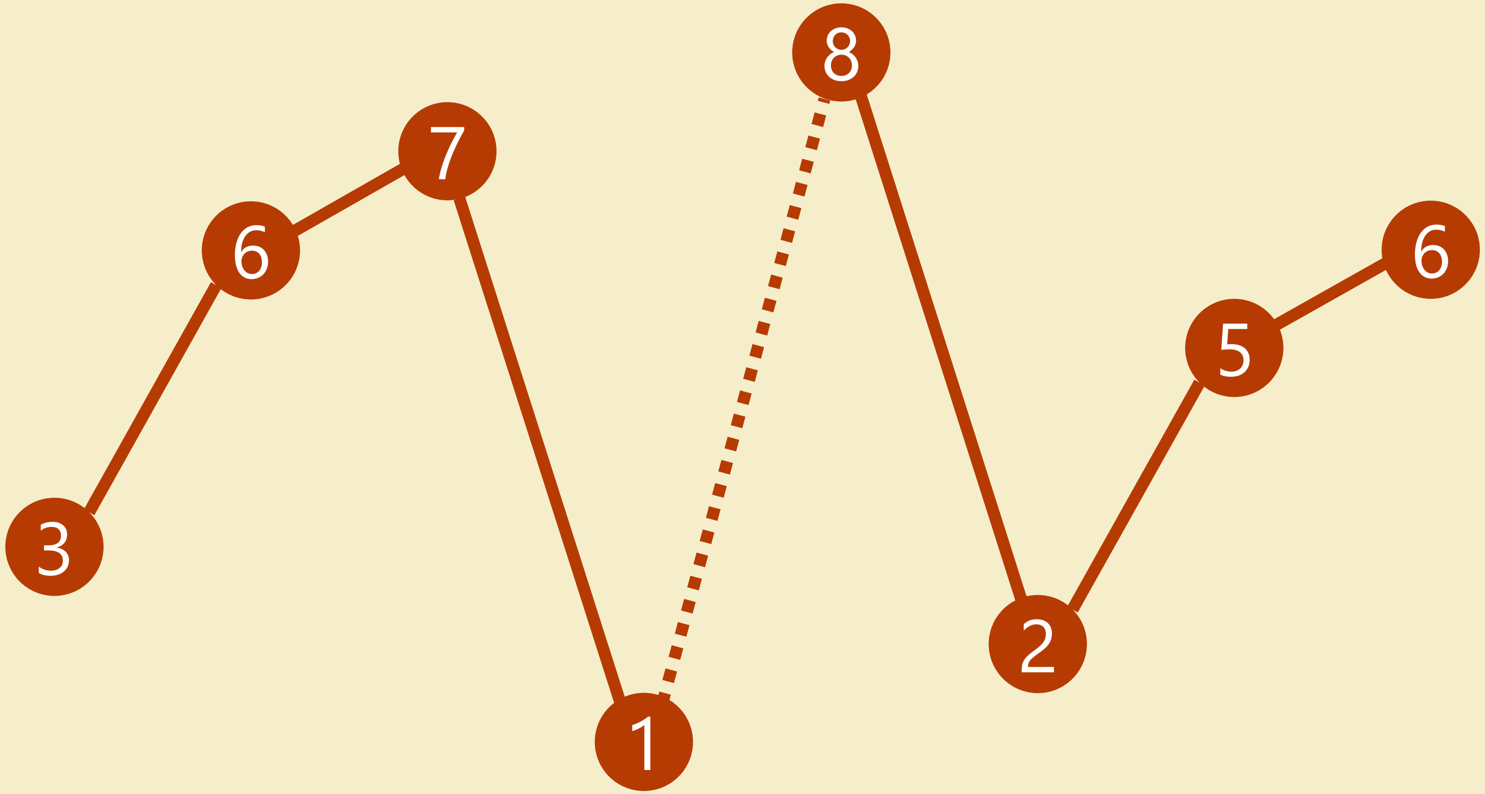
別解

「傾きが最大のを取り除く」というのは、どういうこと？



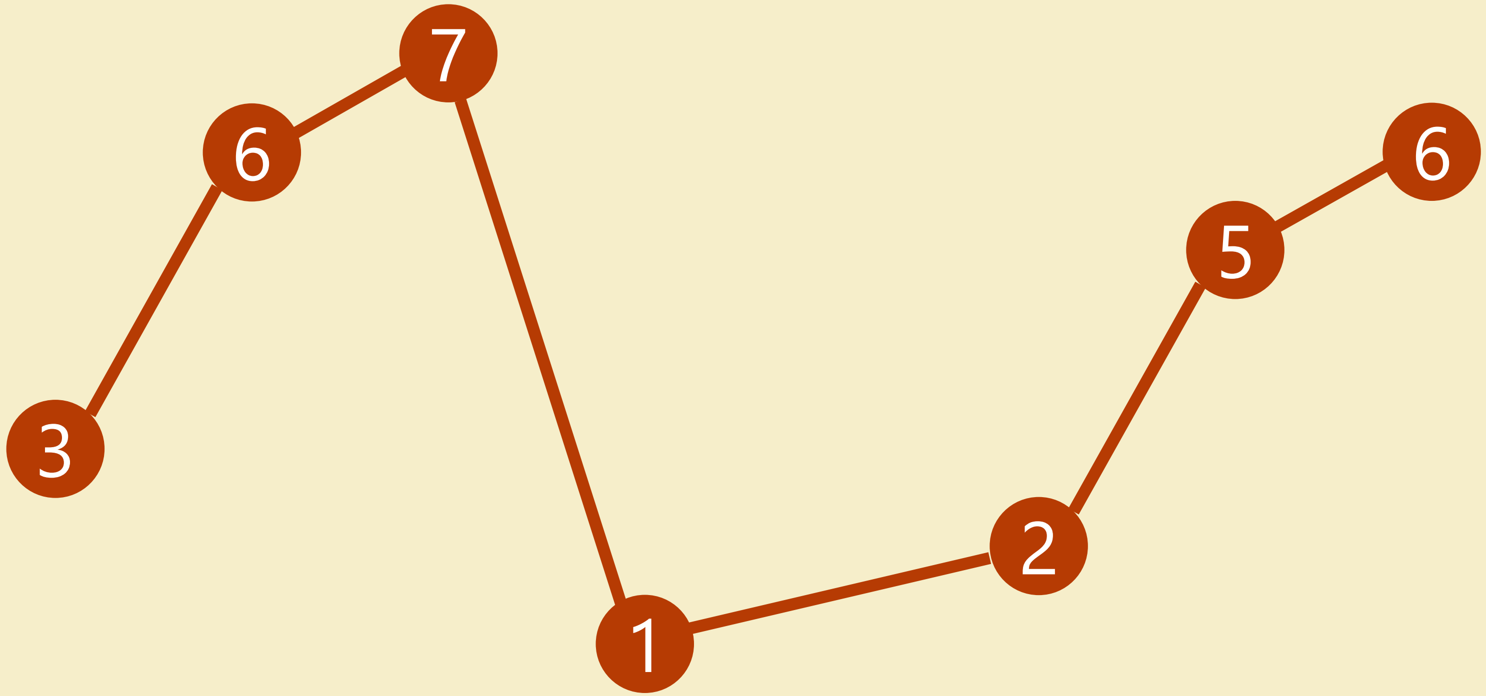
別解

「傾きが最大のを取り除く」というのは、どういうこと？



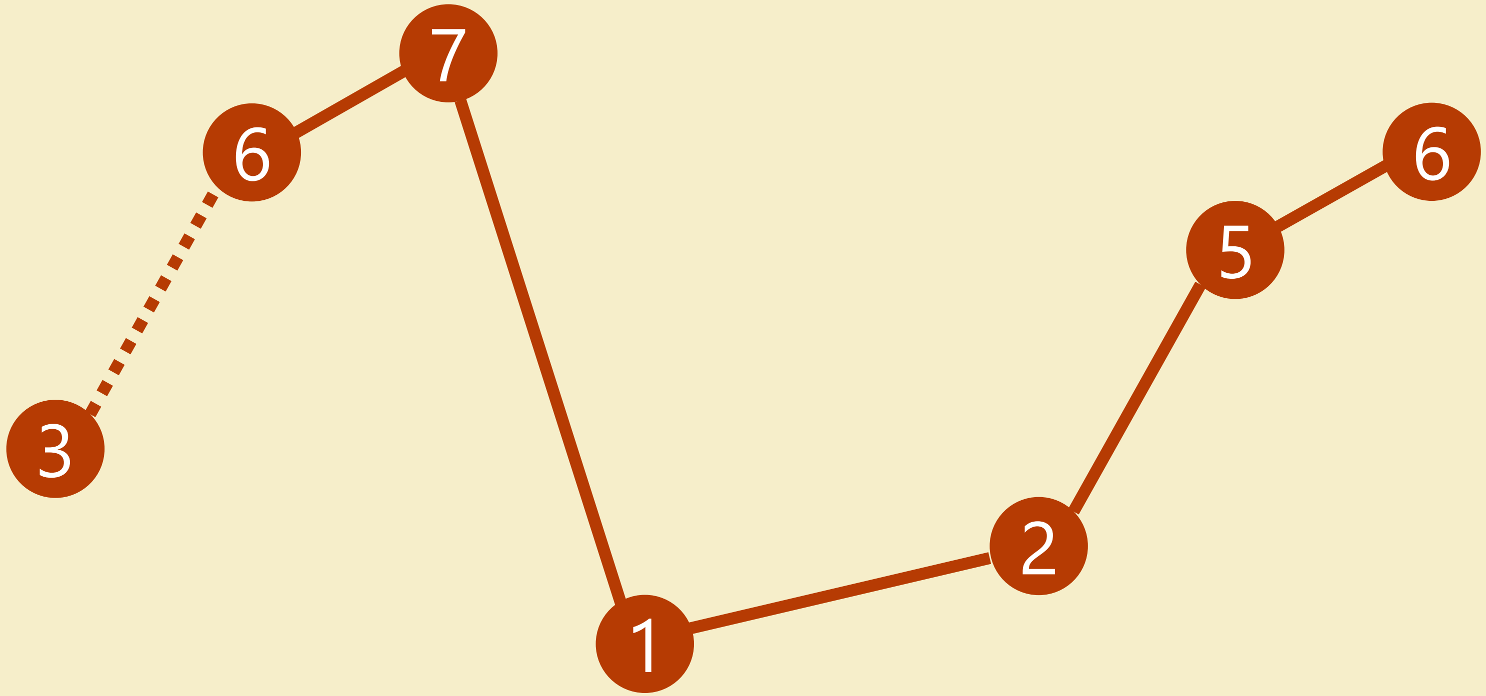
別解

「傾きが最大のを取り除く」というのは、どういうこと？



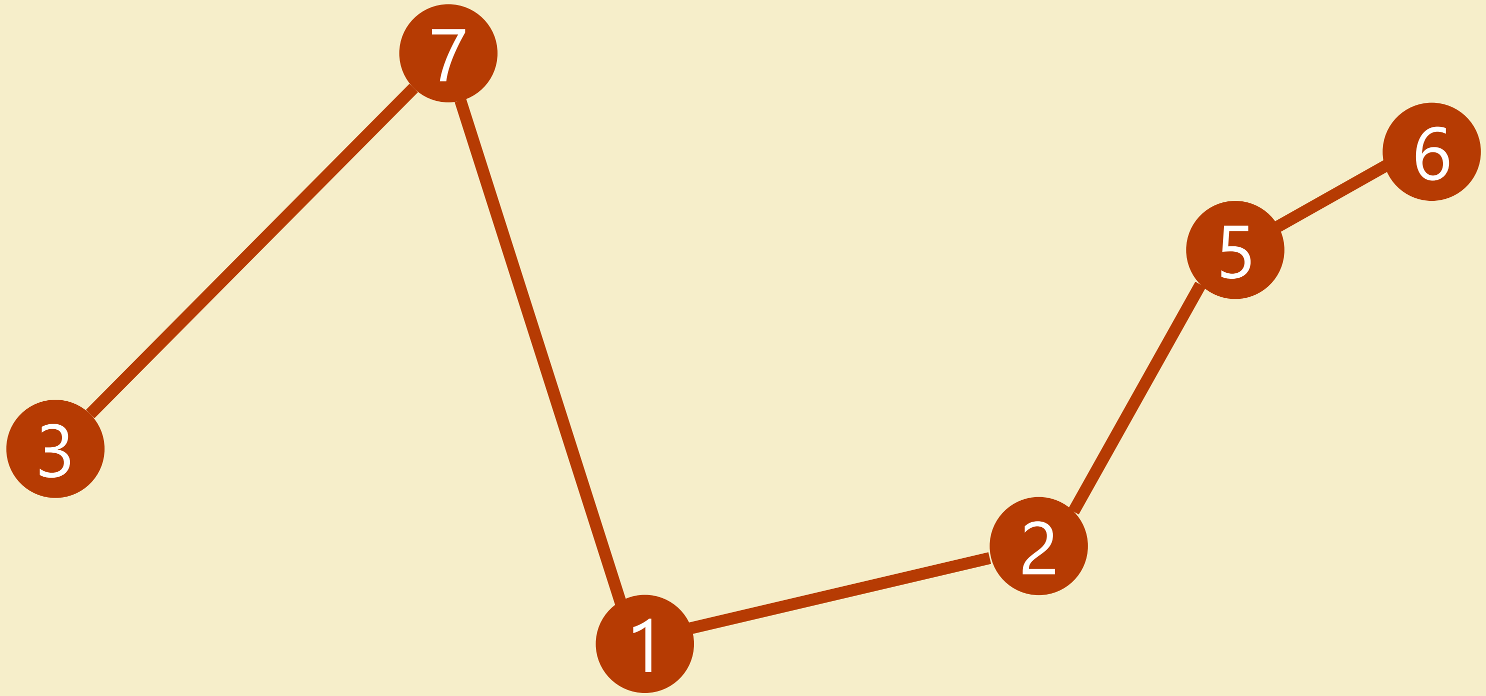
別解

「傾きが最大のを取り除く」というのは、どういうこと？



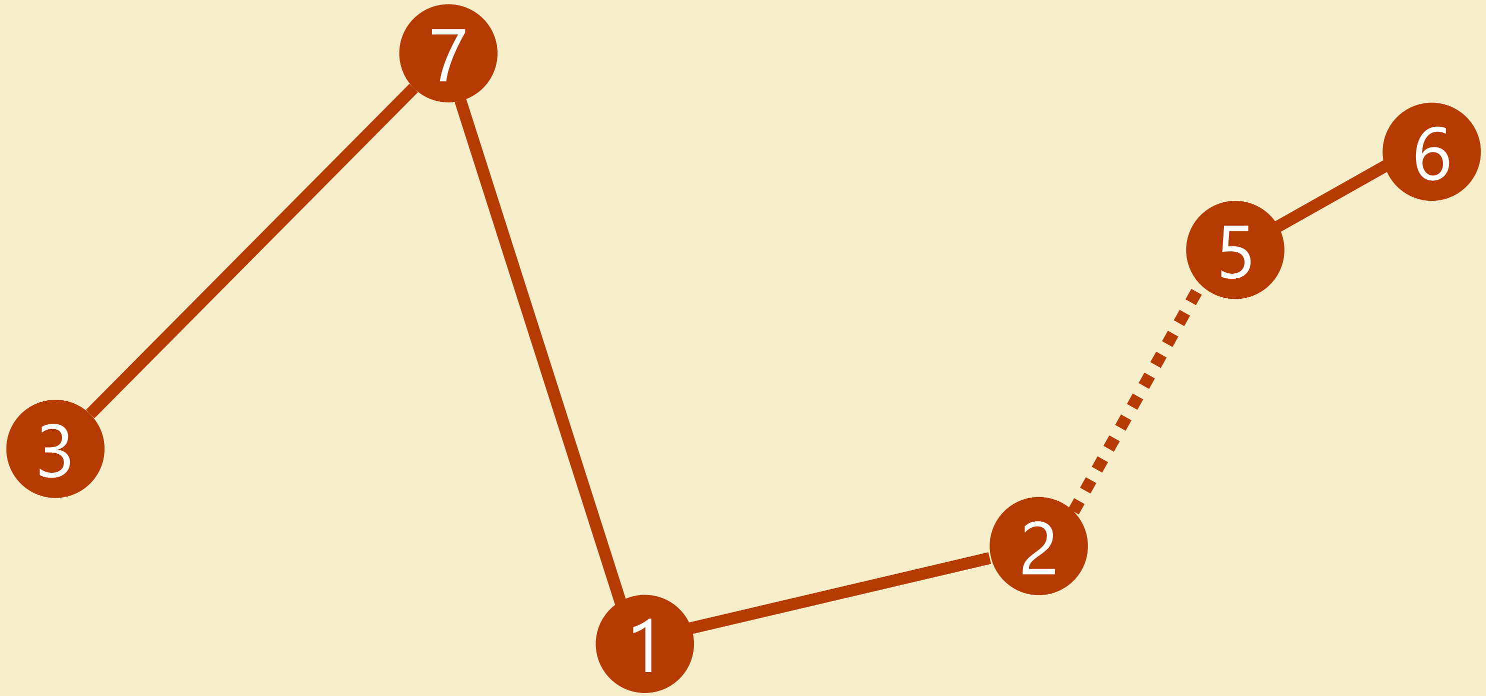
別解

「傾きが最大のを取り除く」というのは、どういうこと？



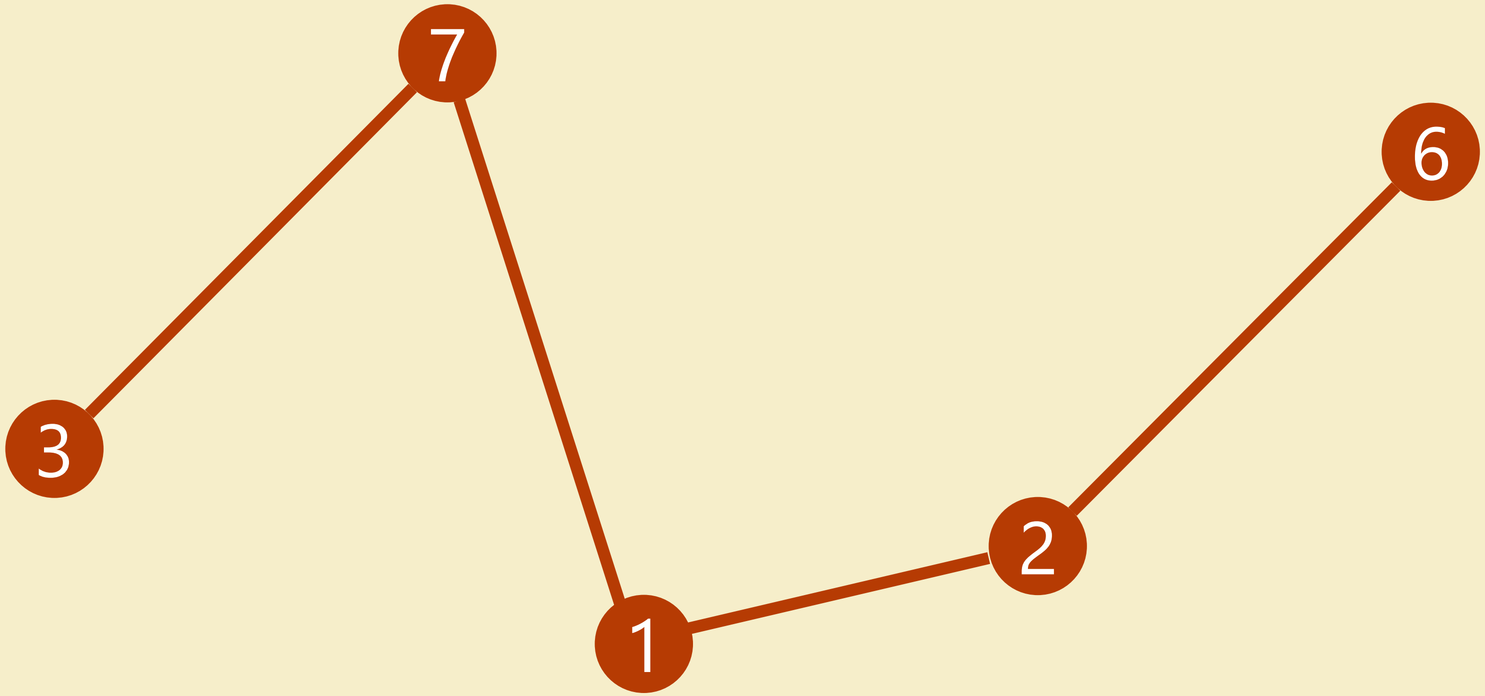
別解

「傾きが最大のを取り除く」というのは、どういうこと？



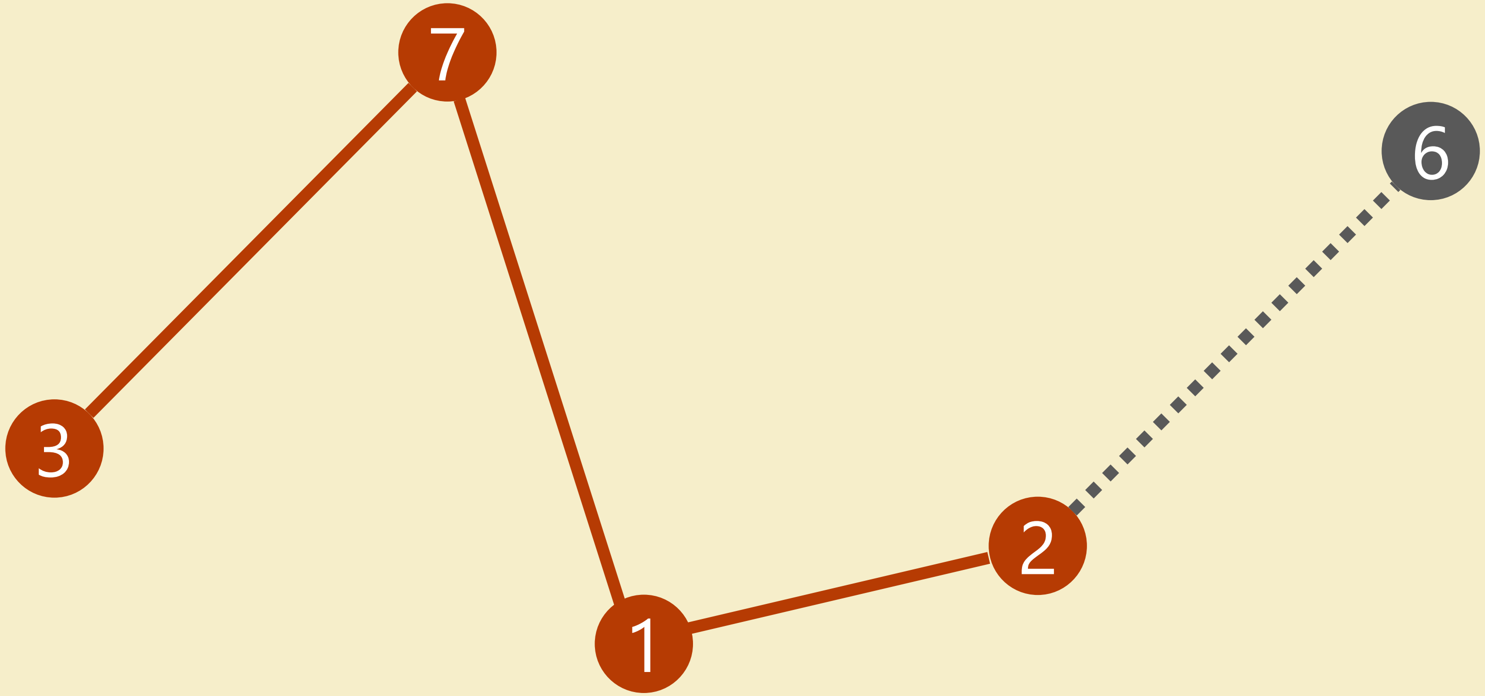
別解

「傾きが最大のを取り除く」というのは、どういうこと？



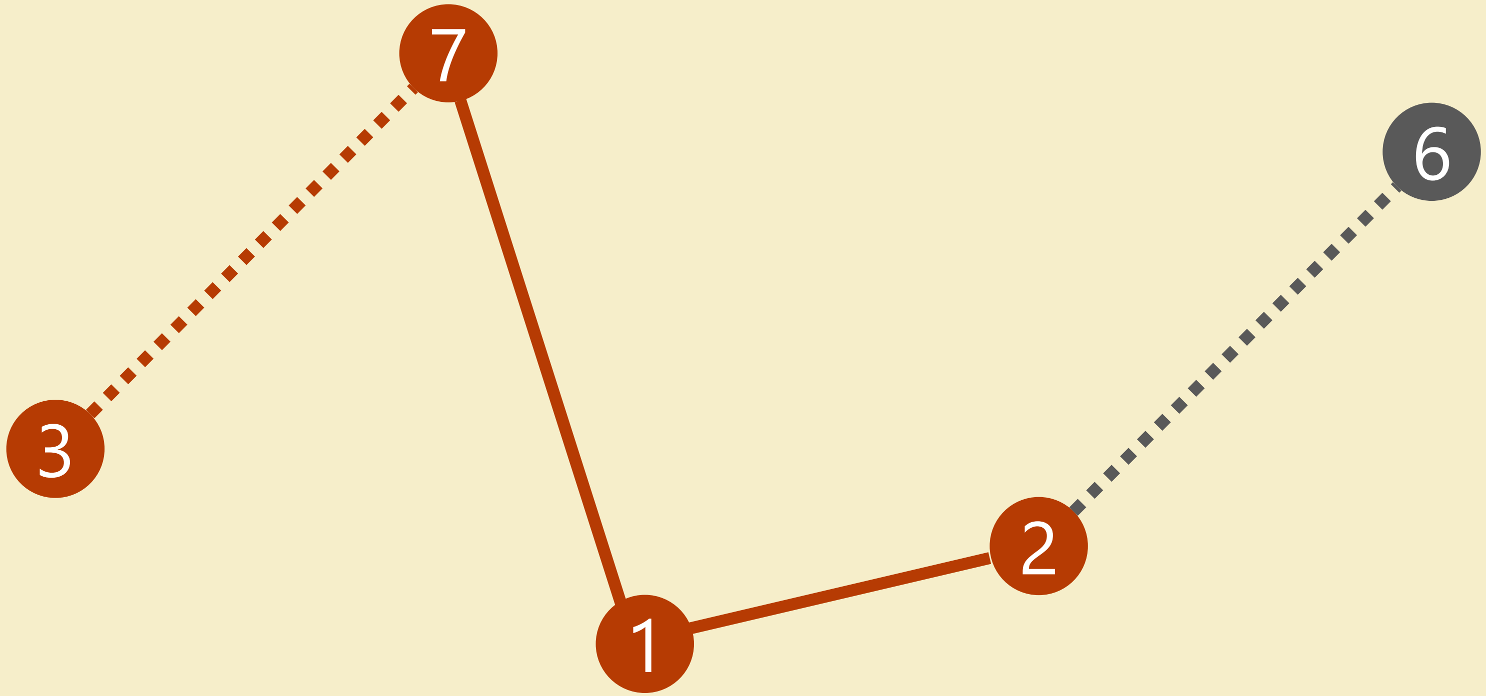
別解

「傾きが最大のを取り除く」というのは、どういうこと？



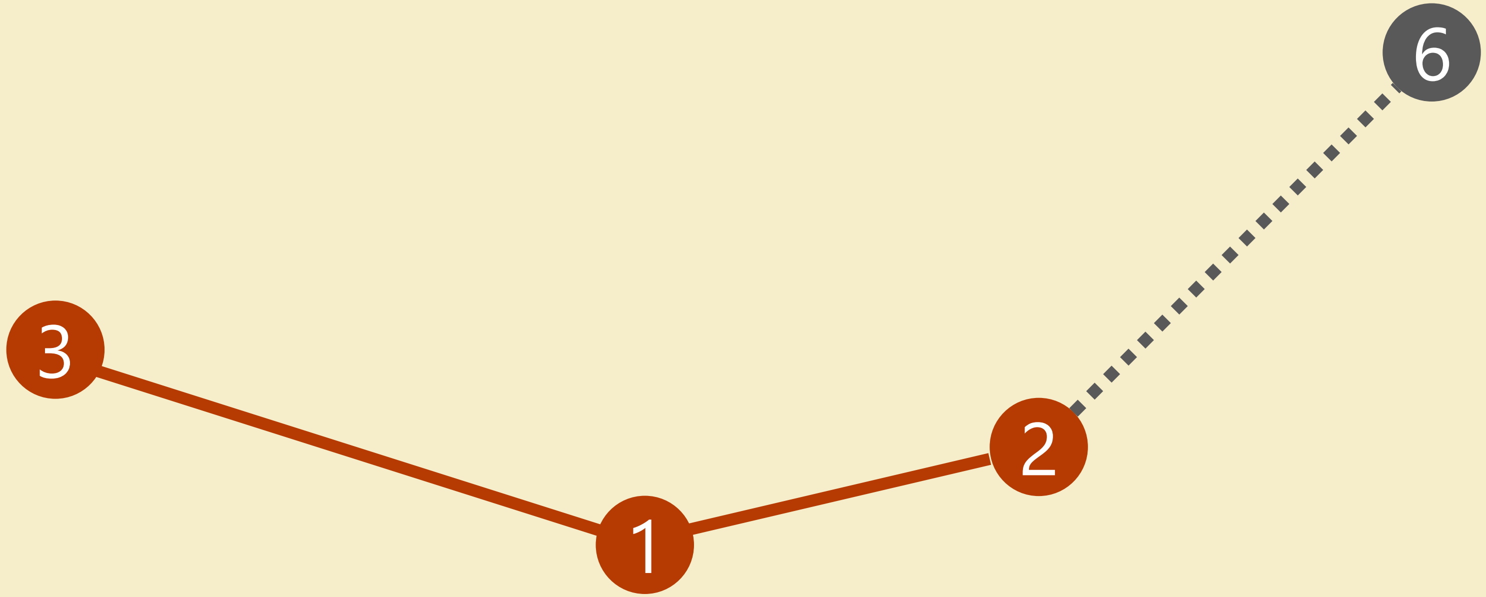
別解

「傾きが最大のを取り除く」というのは、どういうこと？



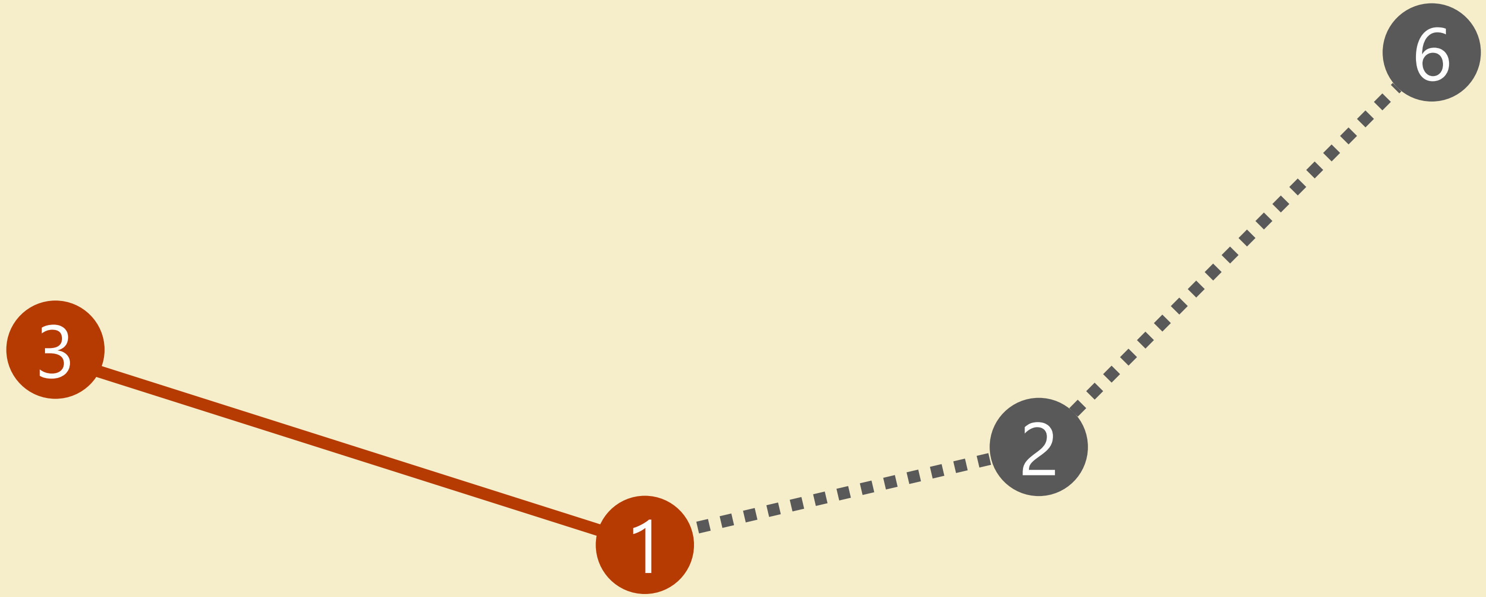
別解

「傾きが最大のを取り除く」というのは、どういうこと？



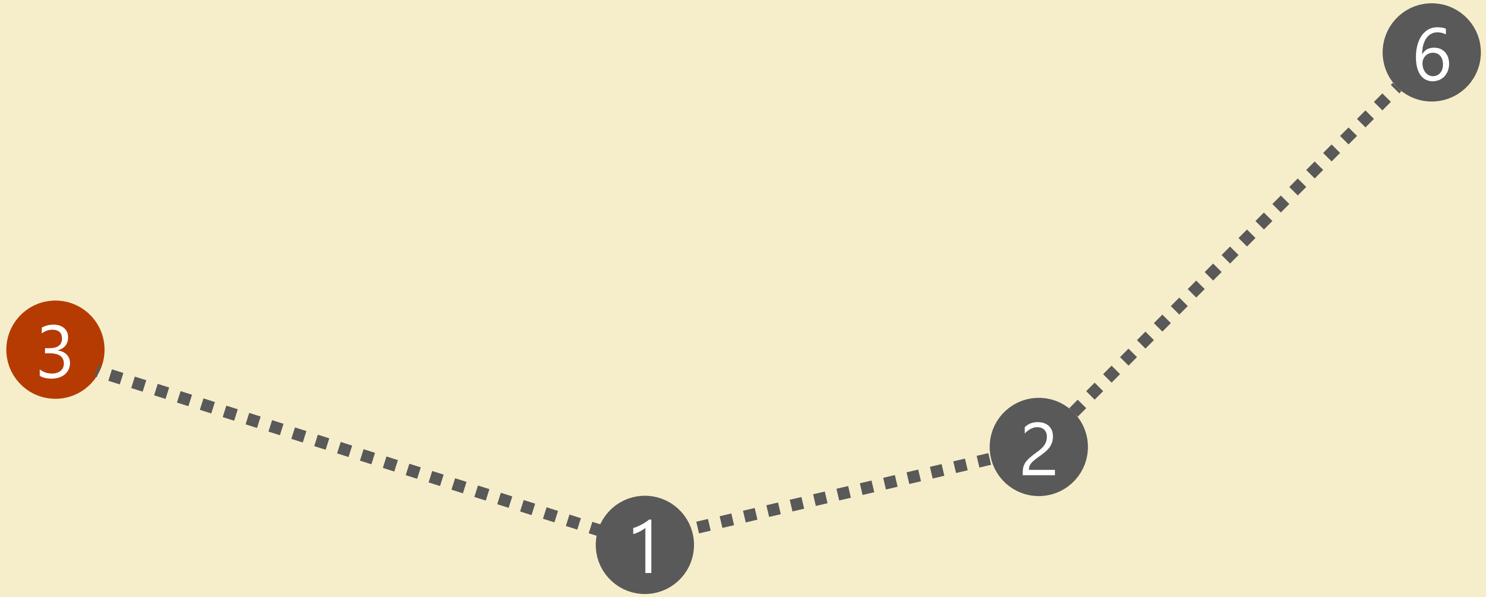
別解

「傾きが最大のを取り除く」というのは、どういうこと?



別解

「傾きが最大のを取り除く」というのは、どういうこと？

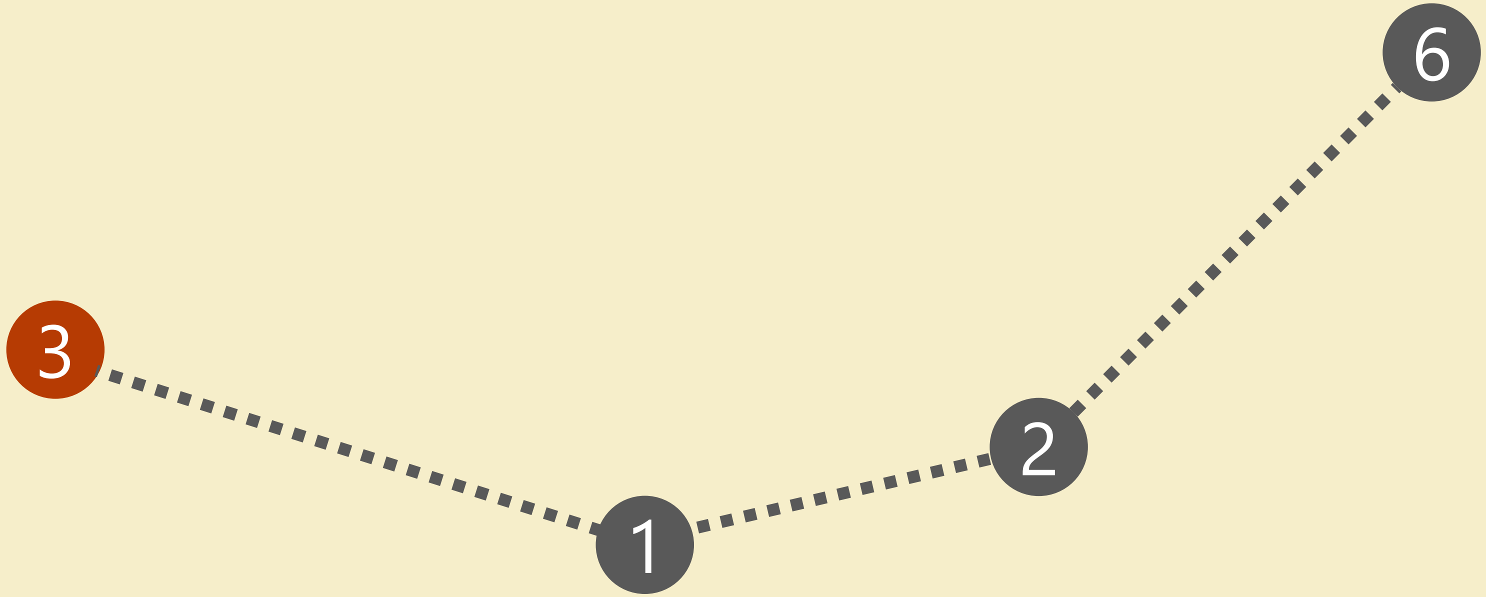


別解

「傾きが最大のを取り除く」というのは、どういうこと？

→ 結局は「点 (i, A_i) の凸包」を考えればよい！

「点 (i, A_i) の凸包」に含まれる道だけに削れる！

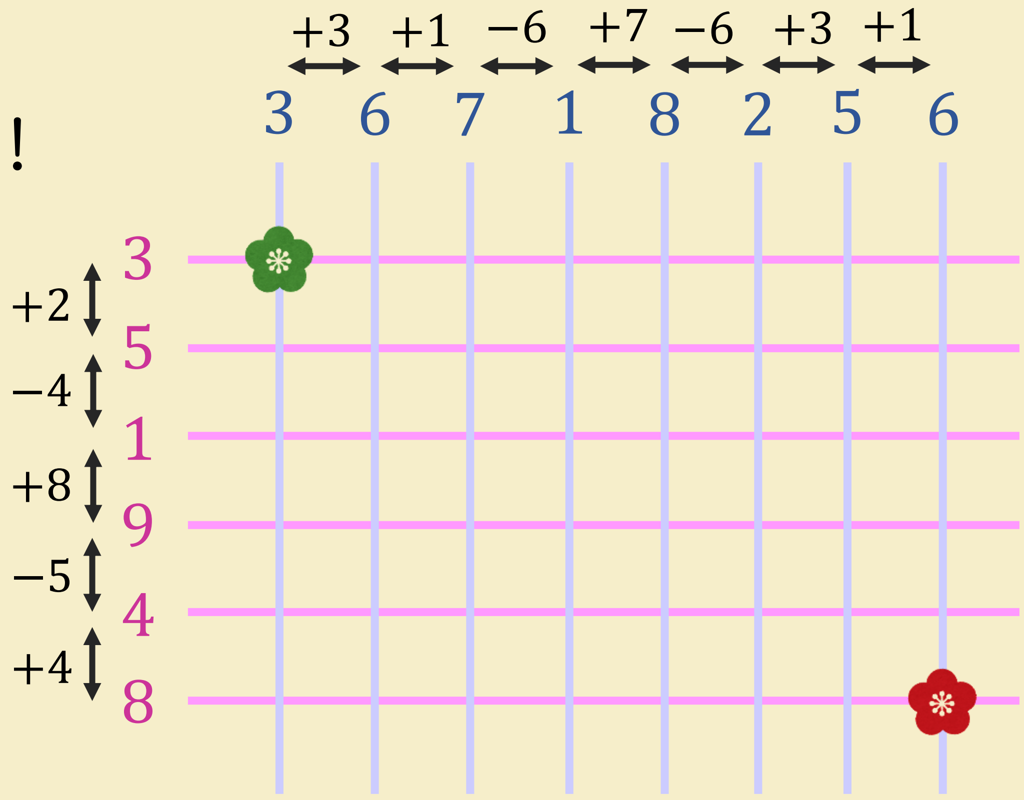


別解

「傾きが最大のを取り除く」というのは、どういうこと？

→ 結局は「点 (i, A_i) の凸包」を考えればよい！

「点 (i, A_i) の凸包」に含まれる道だけに削れる！



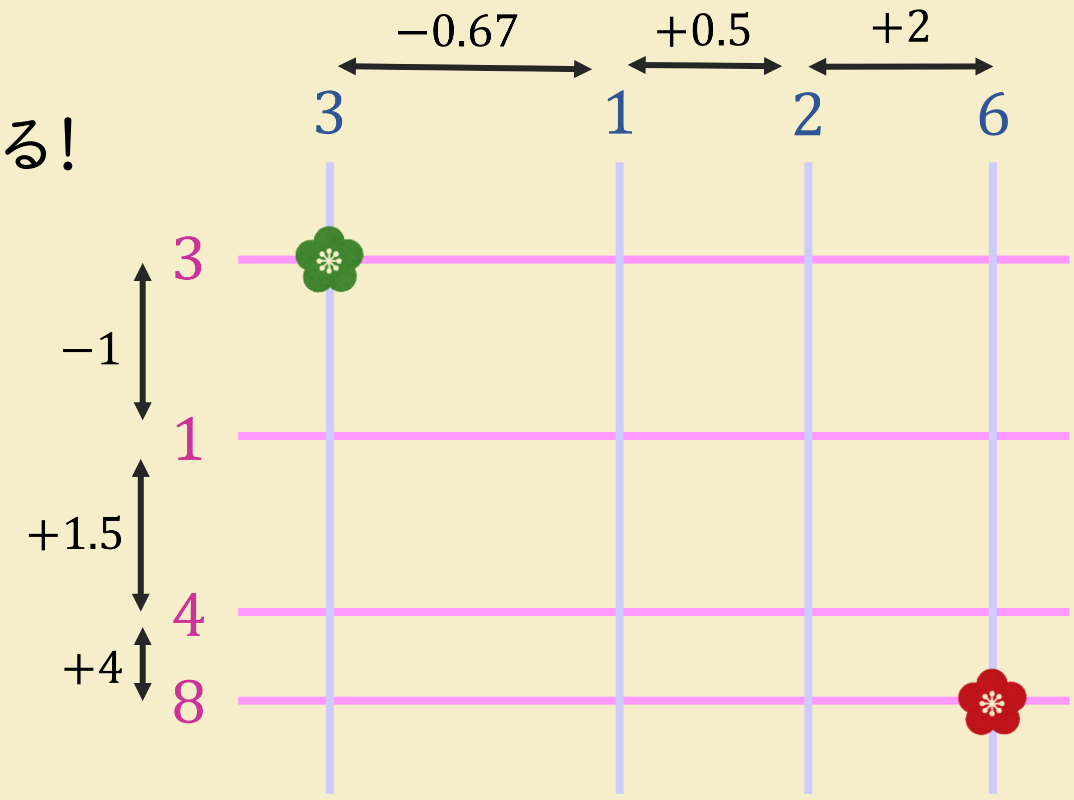
別解

「傾きが最大のを取り除く」というのは、どういうこと？

→ 結局は「点 (i, A_i) の凸包」を考えればよい！

「点 (i, A_i) の凸包」に含まれる道だけに削れる！

これでシミュレーションするとどうなる？



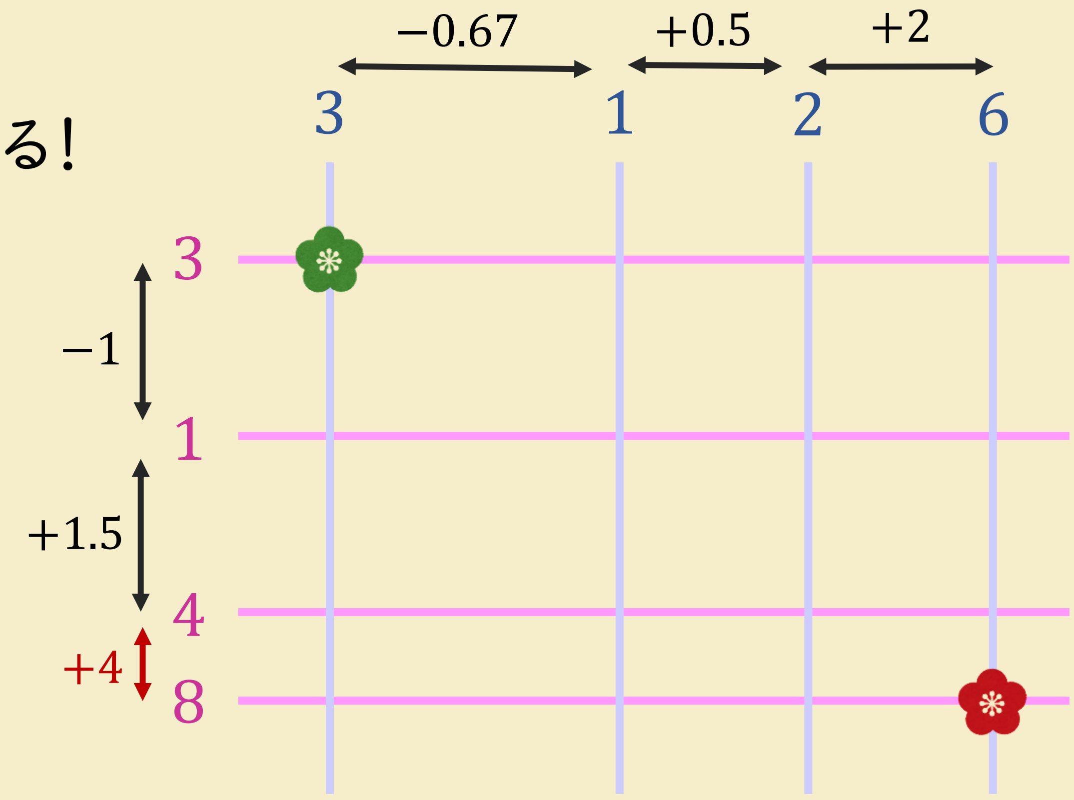
別解

「傾きが最大のを取り除く」というのは、どういうこと？

→ 結局は「点 (i, A_i) の凸包」を考えればよい！

「点 (i, A_i) の凸包」に含まれる道だけに削れる！

これでシミュレーションするとどうなる？

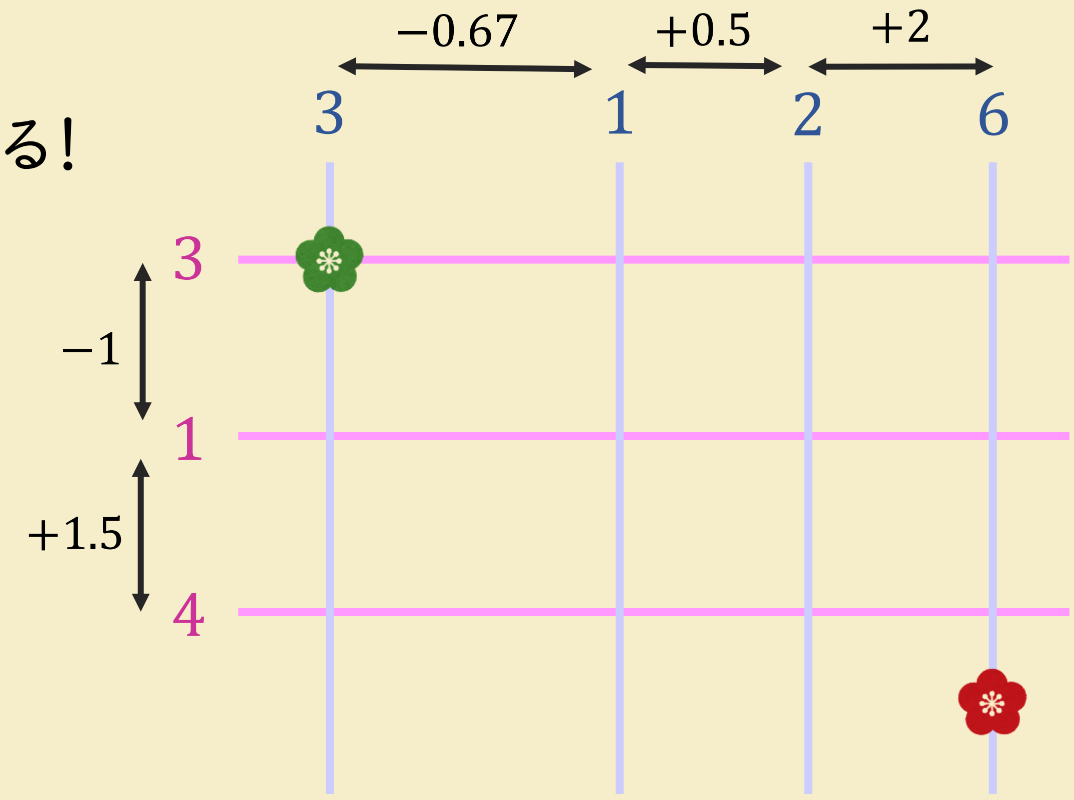


別解

「傾きが最大のを取り除く」というのは、どういうこと？

→ 結局は「点 (i, A_i) の凸包」を考えればよい！
「点 (i, A_i) の凸包」に含まれる道だけに削れる！

これでシミュレーションするとどうなる？



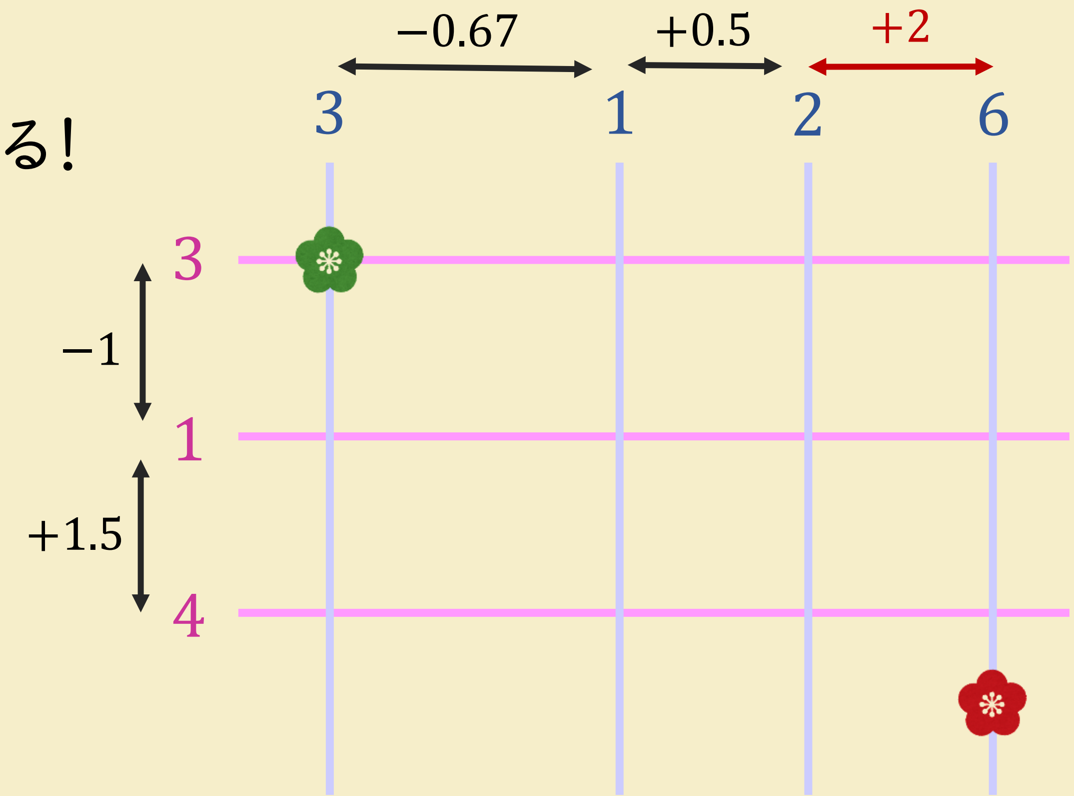
別解

「傾きが最大のを取り除く」というのは、どういうこと？

→ 結局は「点 (i, A_i) の凸包」を考えればよい！

「点 (i, A_i) の凸包」に含まれる道だけに削れる！

これでシミュレーションするとどうなる？



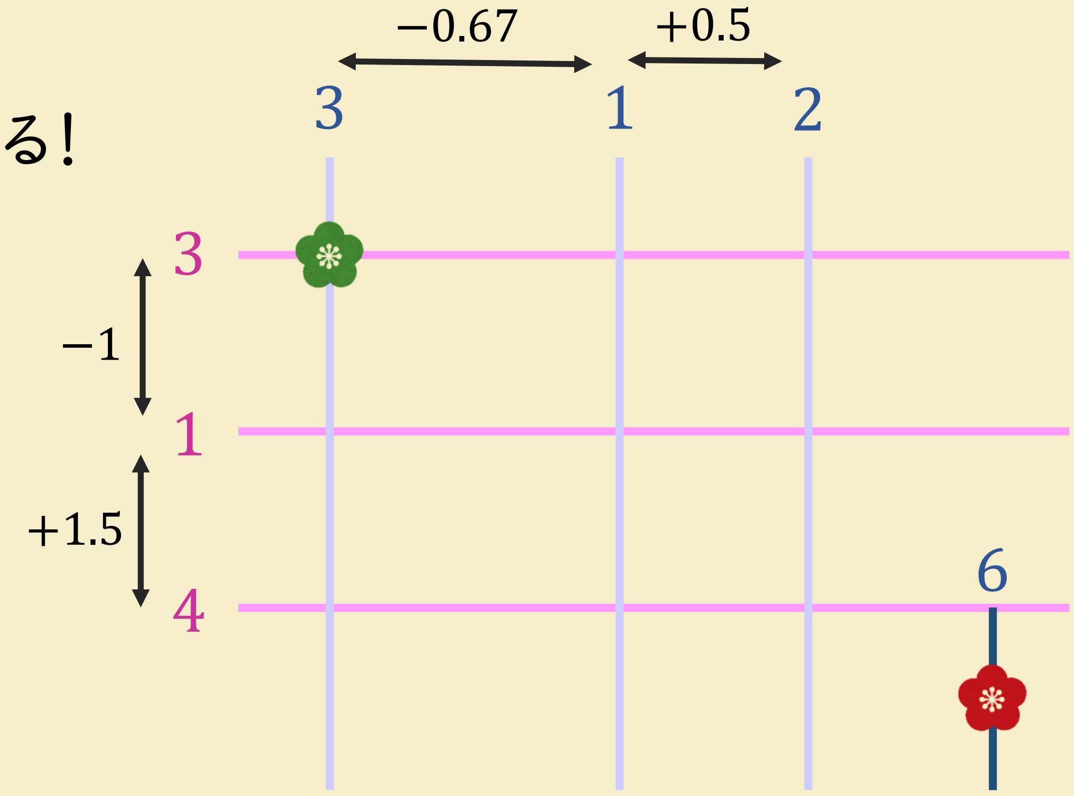
別解

「傾きが最大のを取り除く」というのは、どういうこと？

→ 結局は「点 (i, A_i) の凸包」を考えればよい！

「点 (i, A_i) の凸包」に含まれる道だけに削れる！

これでシミュレーションするとどうなる？



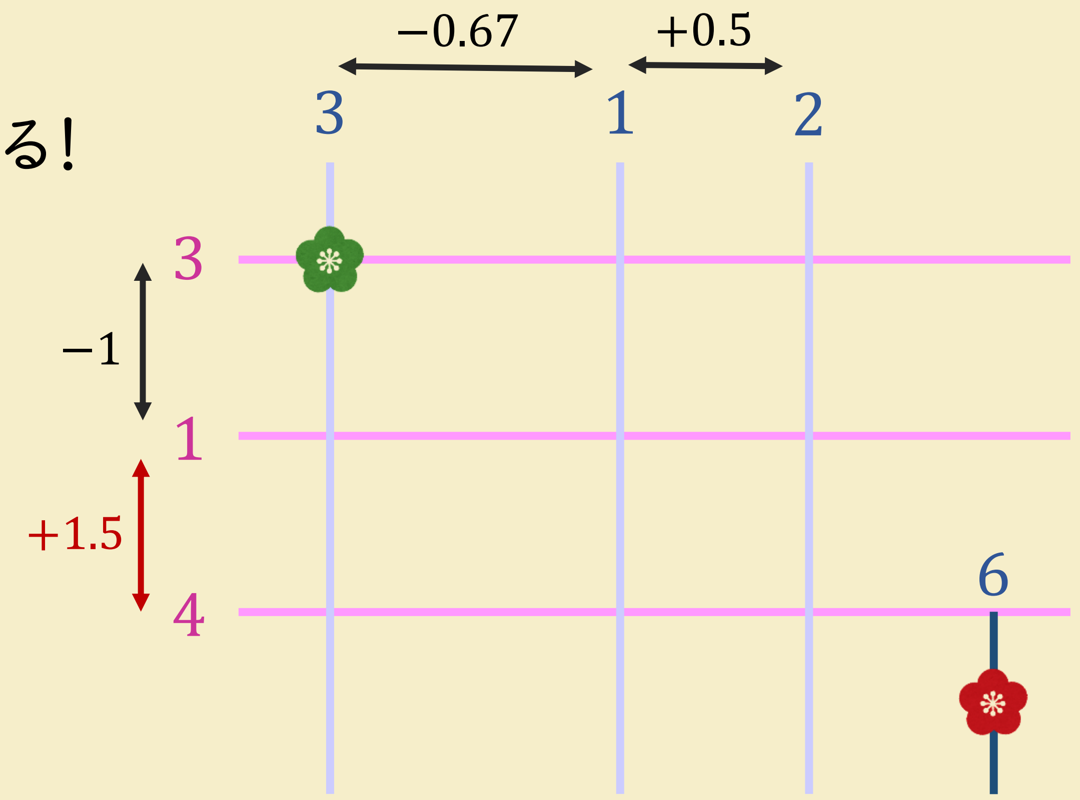
別解

「傾きが最大のを取り除く」というのは、どういうこと？

→ 結局は「点 (i, A_i) の凸包」を考えればよい！

「点 (i, A_i) の凸包」に含まれる道だけに削れる！

これでシミュレーションするとどうなる？



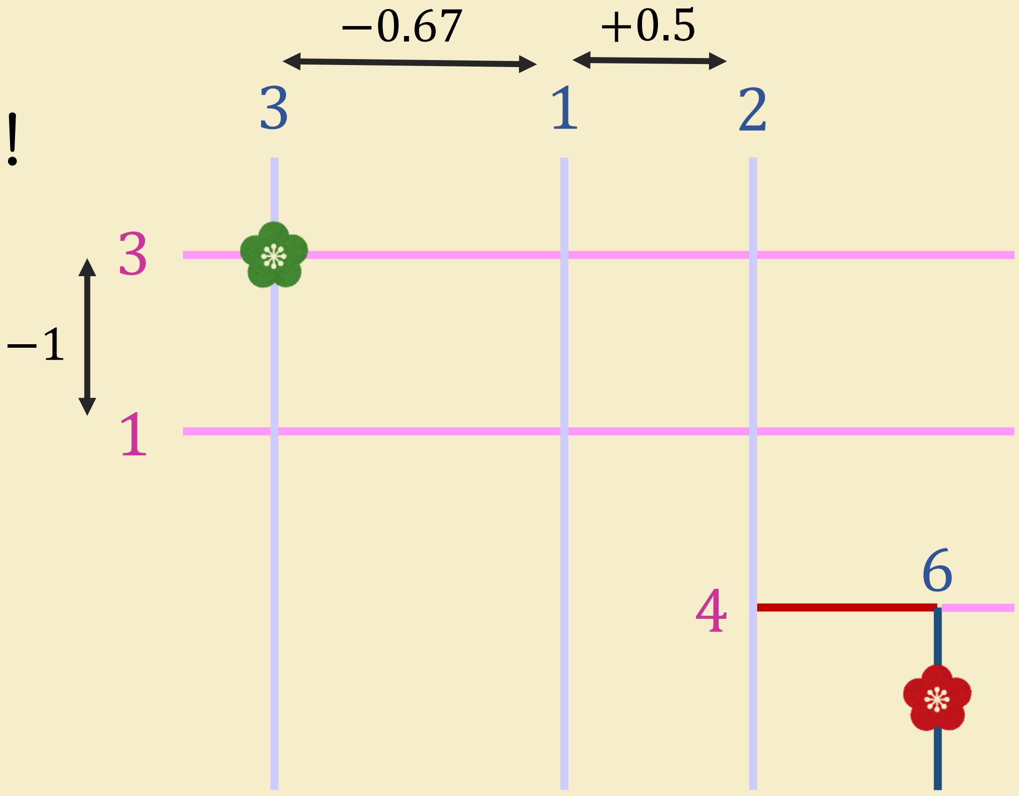
別解

「傾きが最大のを取り除く」というのは、どういうこと？

→ 結局は「点 (i, A_i) の凸包」を考えればよい！

「点 (i, A_i) の凸包」に含まれる道だけに削れる！

これでシミュレーションするとどうなる？



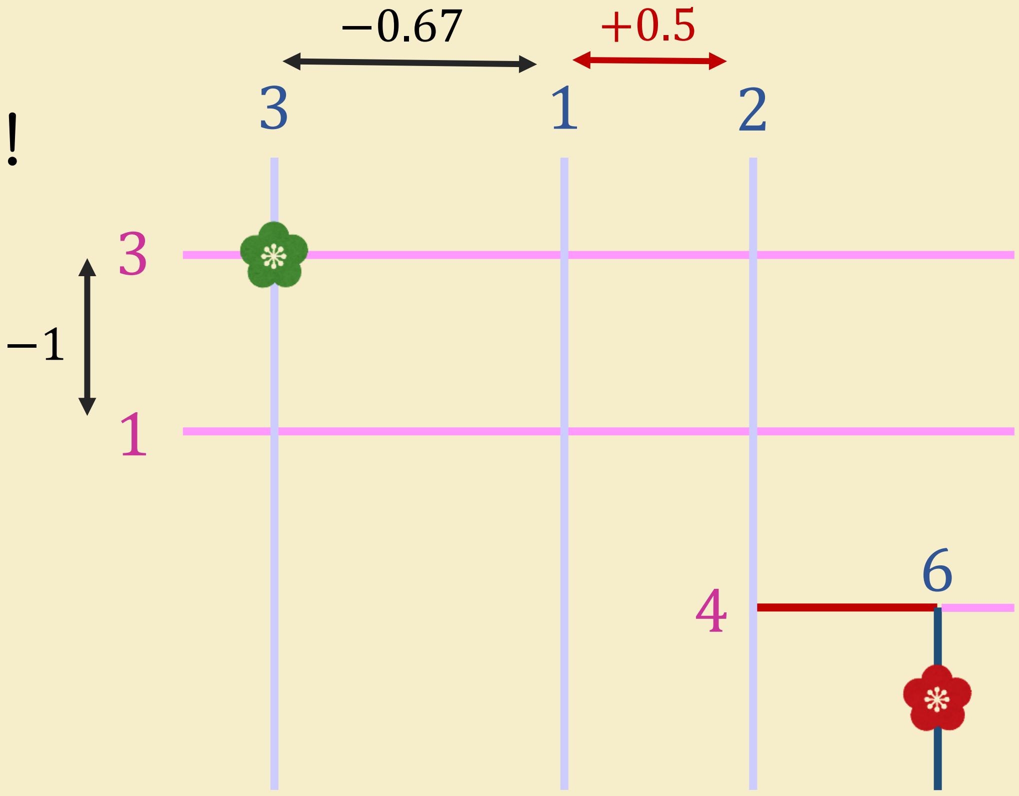
別解

「傾きが最大のを取り除く」というのは、どういうこと？

→ 結局は「点 (i, A_i) の凸包」を考えればよい！

「点 (i, A_i) の凸包」に含まれる道だけに削れる！

これでシミュレーションするとどうなる？



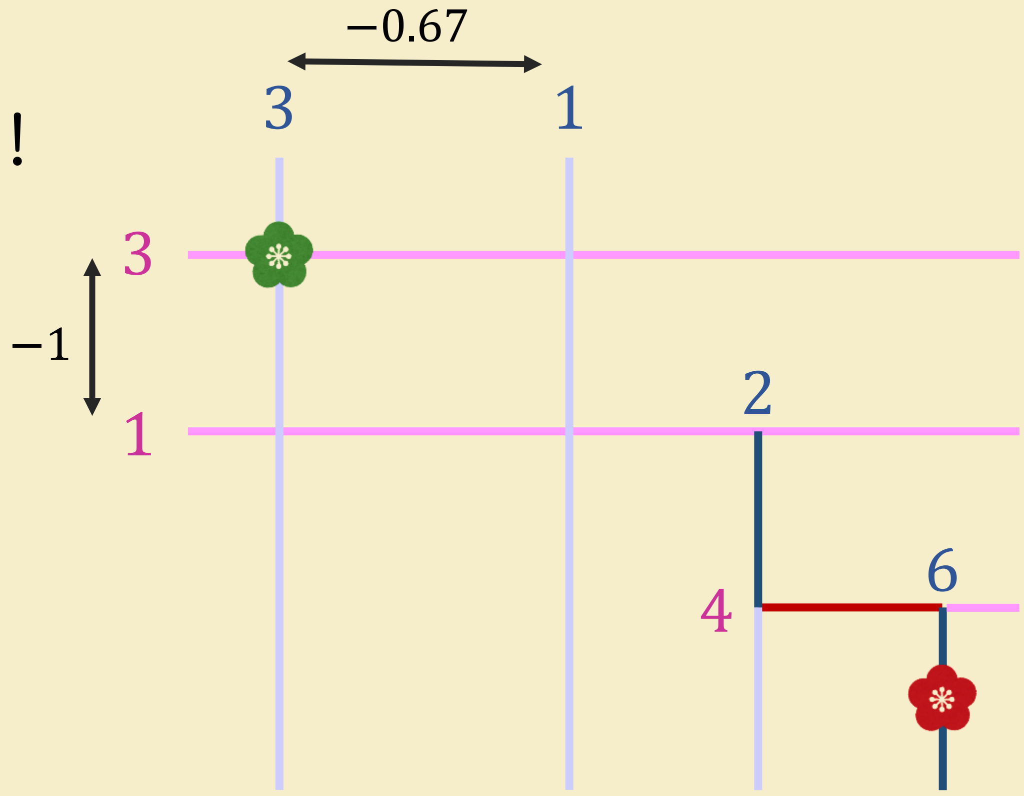
別解

「傾きが最大のを取り除く」というのは、どういうこと？

→ 結局は「点 (i, A_i) の凸包」を考えればよい！

「点 (i, A_i) の凸包」に含まれる道だけに削れる！

これでシミュレーションするとどうなる？



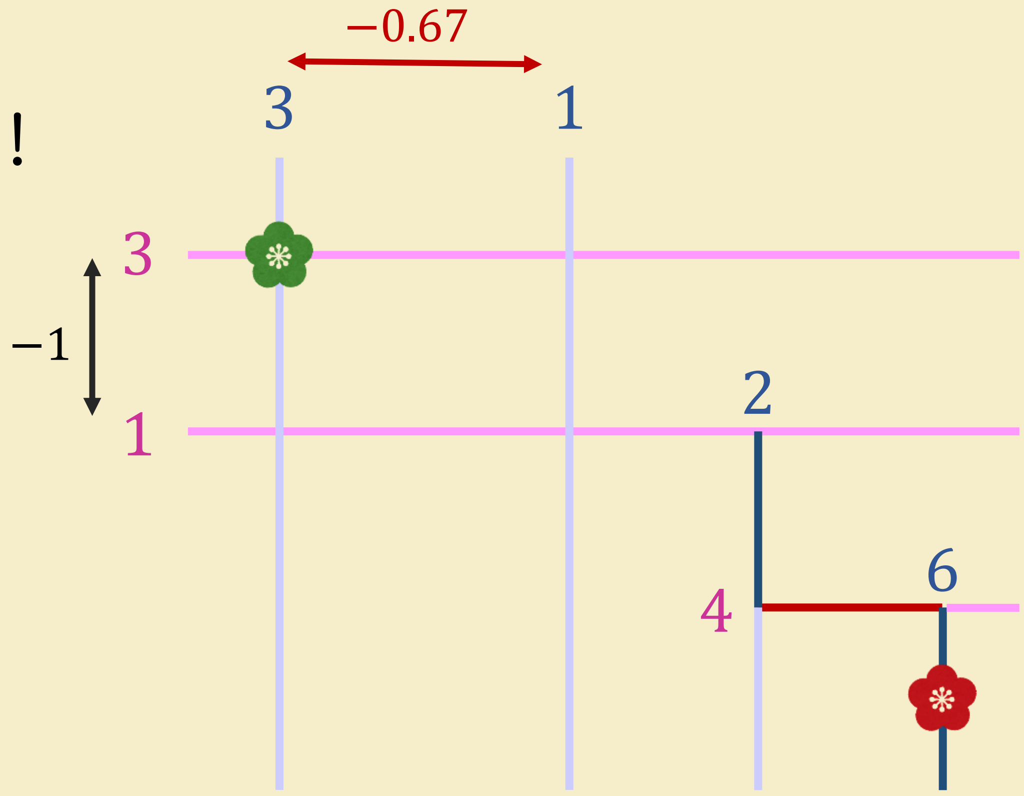
別解

「傾きが最大のを取り除く」というのは、どういうこと？

→ 結局は「点 (i, A_i) の凸包」を考えればよい！

「点 (i, A_i) の凸包」に含まれる道だけに削れる！

これでシミュレーションするとどうなる？



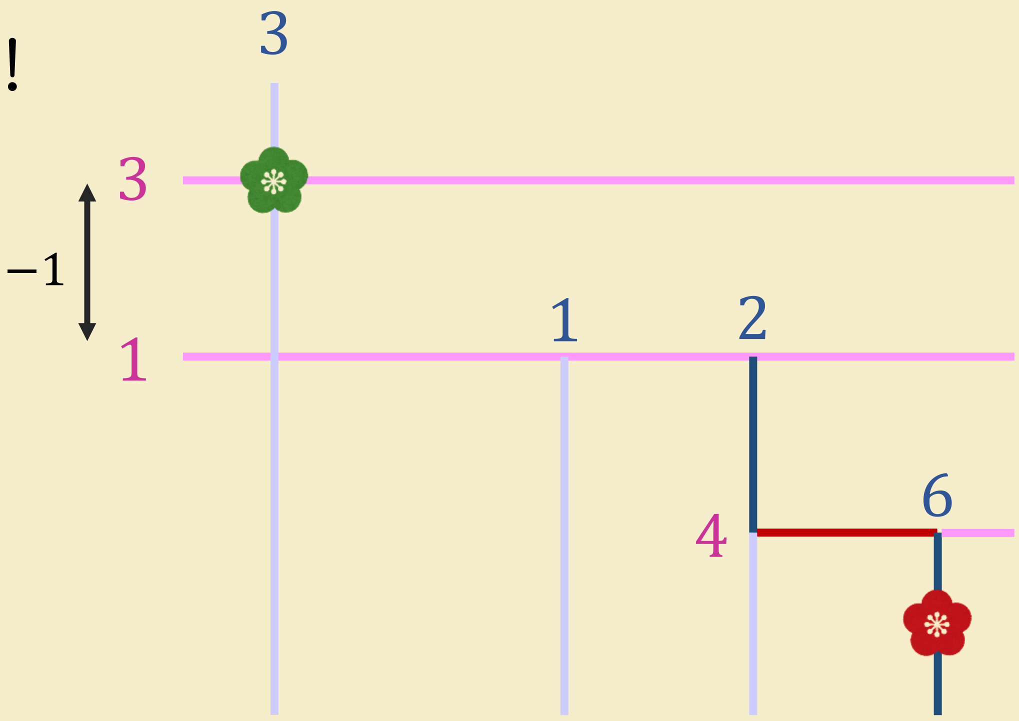
別解

「傾きが最大のを取り除く」というのは、どういうこと？

→ 結局は「点 (i, A_i) の凸包」を考えればよい！

「点 (i, A_i) の凸包」に含まれる道だけに削れる！

これでシミュレーションするとどうなる？



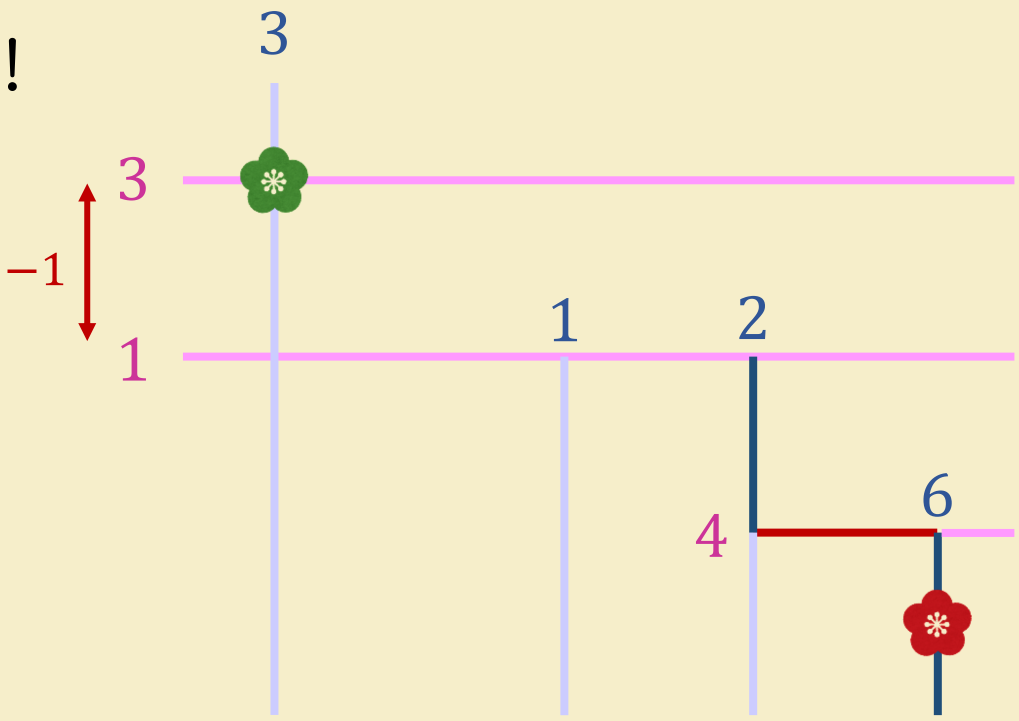
別解

「傾きが最大のを取り除く」というのは、どういうこと？

→ 結局は「点 (i, A_i) の凸包」を考えればよい！

「点 (i, A_i) の凸包」に含まれる道だけに削れる！

これでシミュレーションするとどうなる？



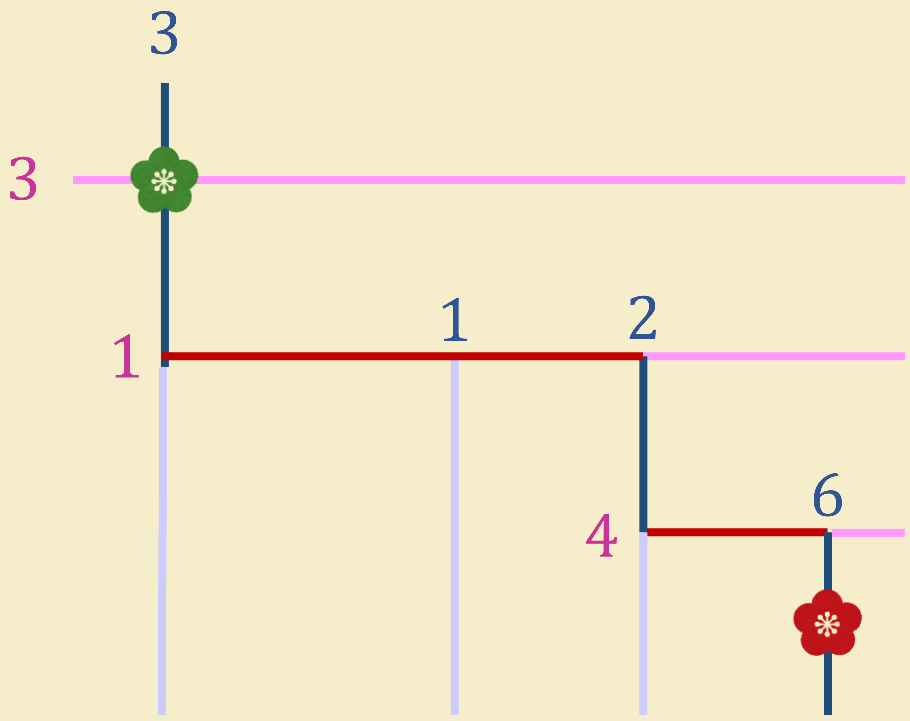
別解

「傾きが最大のを取り除く」というのは、どういうこと？

→ 結局は「点 (i, A_i) の凸包」を考えればよい！

「点 (i, A_i) の凸包」に含まれる道だけに削れる！

これでシミュレーションするとどうなる？



別解

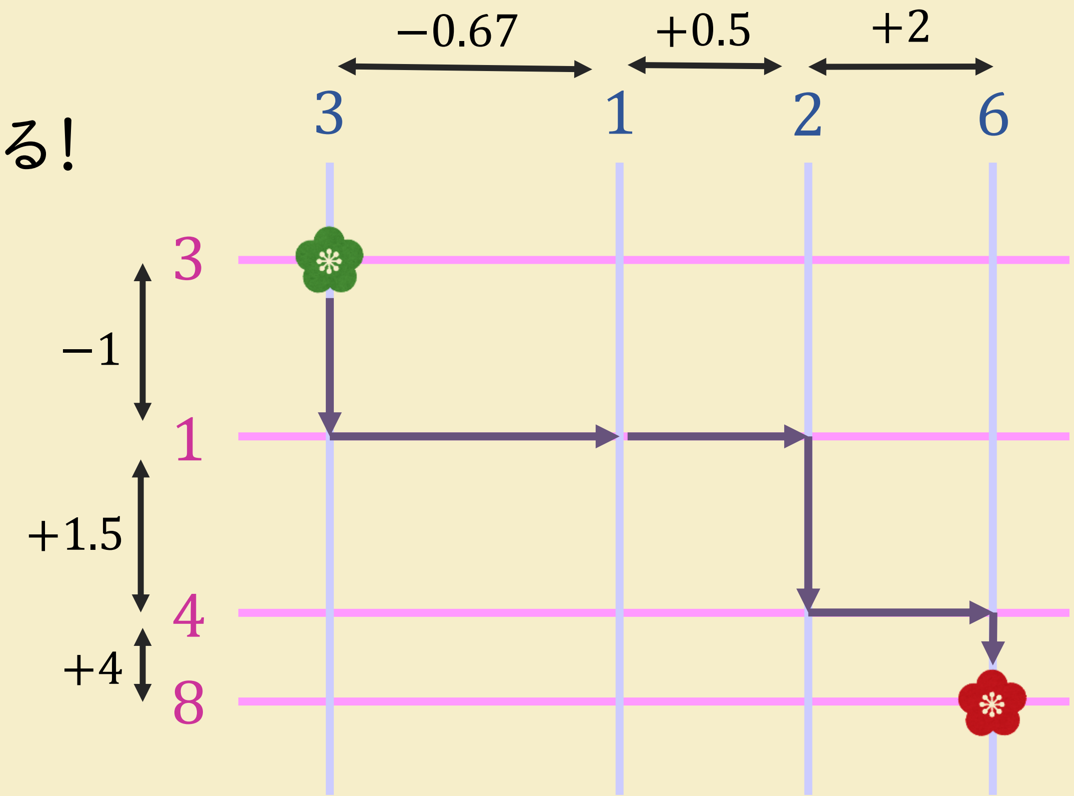
「傾きが最大のを取り除く」というのは、どういうこと？

→ 結局は「点 (i, A_i) の凸包」を考えればよい！

「点 (i, A_i) の凸包」に含まれる道だけに削れる！

これでシミュレーションするとどうなる？

よって、答えはこうなります！



別解

「傾きが最大のを取り除く」というのは、どういうこと？

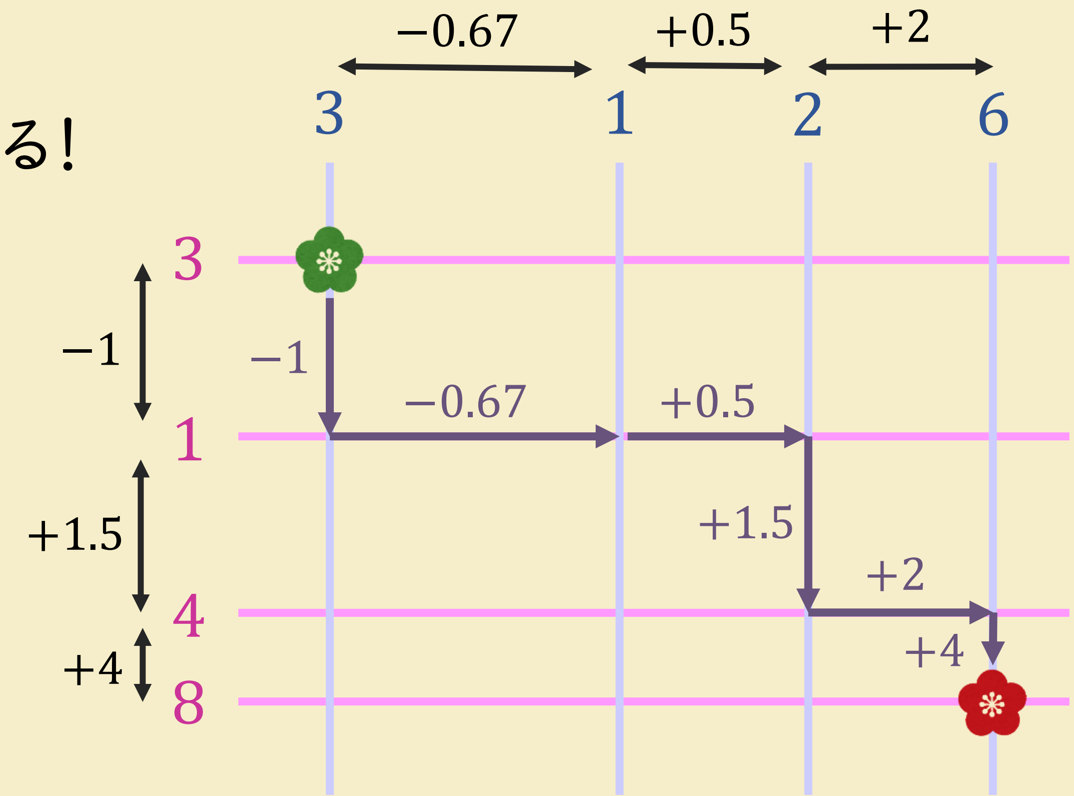
→ 結局は「点 (i, A_i) の凸包」を考えればよい！

「点 (i, A_i) の凸包」に含まれる道だけに削れる！

これでシミュレーションするとどうなる？

よって、答えはこうなります！

「傾きの小さい道路」から通るのが最適！



別解

アルゴリズム

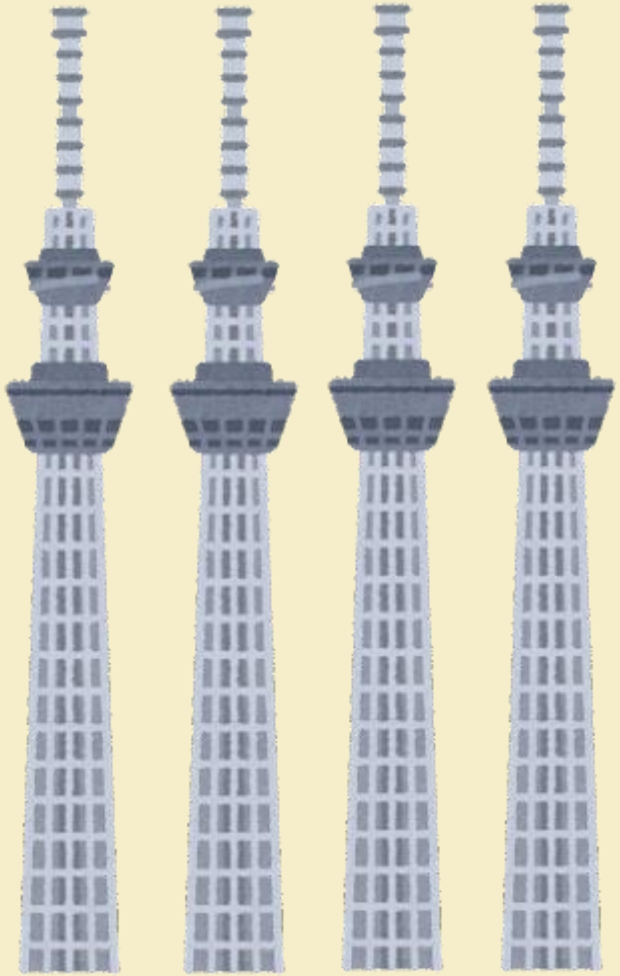
1. 点 (i, A_i) の凸包を求める → 計算量 $O(H)$
2. 点 (j, B_j) の凸包を求める → 計算量 $O(W)$
3. 凸包の傾きをマージする → 計算量 $O(H+W)$

別解

アルゴリズム

1. 点 (i, A_i) の凸包を求める → 計算量 $O(H)$
2. 点 (j, B_j) の凸包を求める → 計算量 $O(W)$
3. 凸包の傾きをマージする → 計算量 $O(H+W)$

全体計算量 $O(H+W)$



得点分布

10 点

24 人

40 点

5 人

100 点

0 人

得点分布

全員 10 点以上

Congratulations!!!