AtCoder Beginner Contest 035 解説

AtCoder株式会社

A: テレビ

問題概要と解法

- 幅 W 高さ H のテレビがある
- 画面のアスペクト比が 4:3 か 16:9 のどちらか判定せよ

- $3W = 4H$ を満たすならば 4:3 そうでないならば 16:9

別解

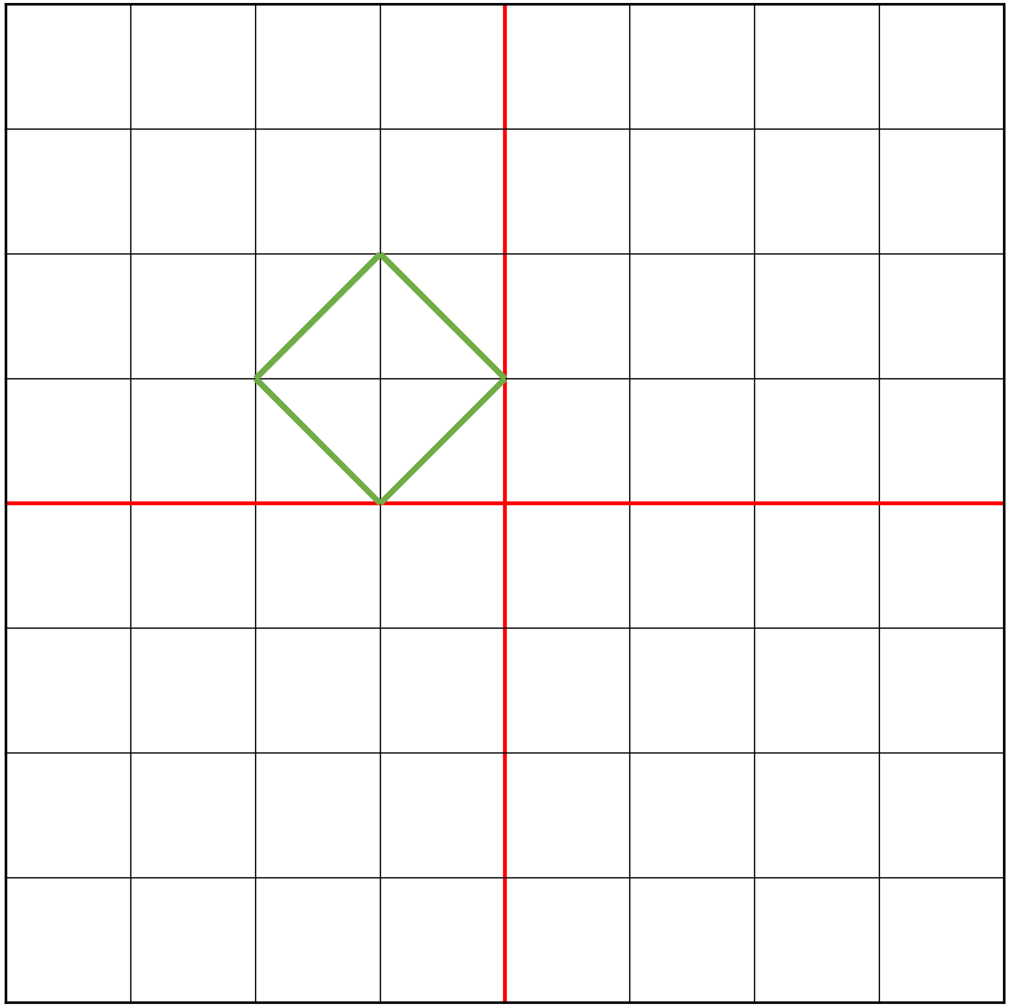
- GCD(最大公約数)を求めて割る
  - 4 と 3 、 16 と 9 は互いに素なので大丈夫
- W と H の両方を割りきれる数を探して割り続ける
- 4:3 であるような画面サイズ全てを全探索
  - 4:3, 8:6, 12:9, …, 100000:75000 にあてはまるか調べる

B: ドローン

問題概要

- ドローンを上下左右の 4 方向に何回か移動させた
- いくつかはどの方向に移動させたか分からなくなった
- ドローンの原点からのマンハッタン距離の最大値、最小値を求めよ

サンプル1: **UL?**



原点から遠ざかると距離は 3 、近づくと距離は 1

考察

- 命令回数は最大 100,000 と多いので、
  移動した方向を全て試すことはできない
  - 最悪ケースで 4^{100000}通りととても多い

- 距離の最大値と最小値だけ分かればよいことに着目
  - 最終的な位置が原点から遠くなる、あるいは
    原点に近くなる方向にだけ移動することを考えればよい

距離の最大値(100 点)

- 最終的な位置が原点から遠ざかるように移動したい

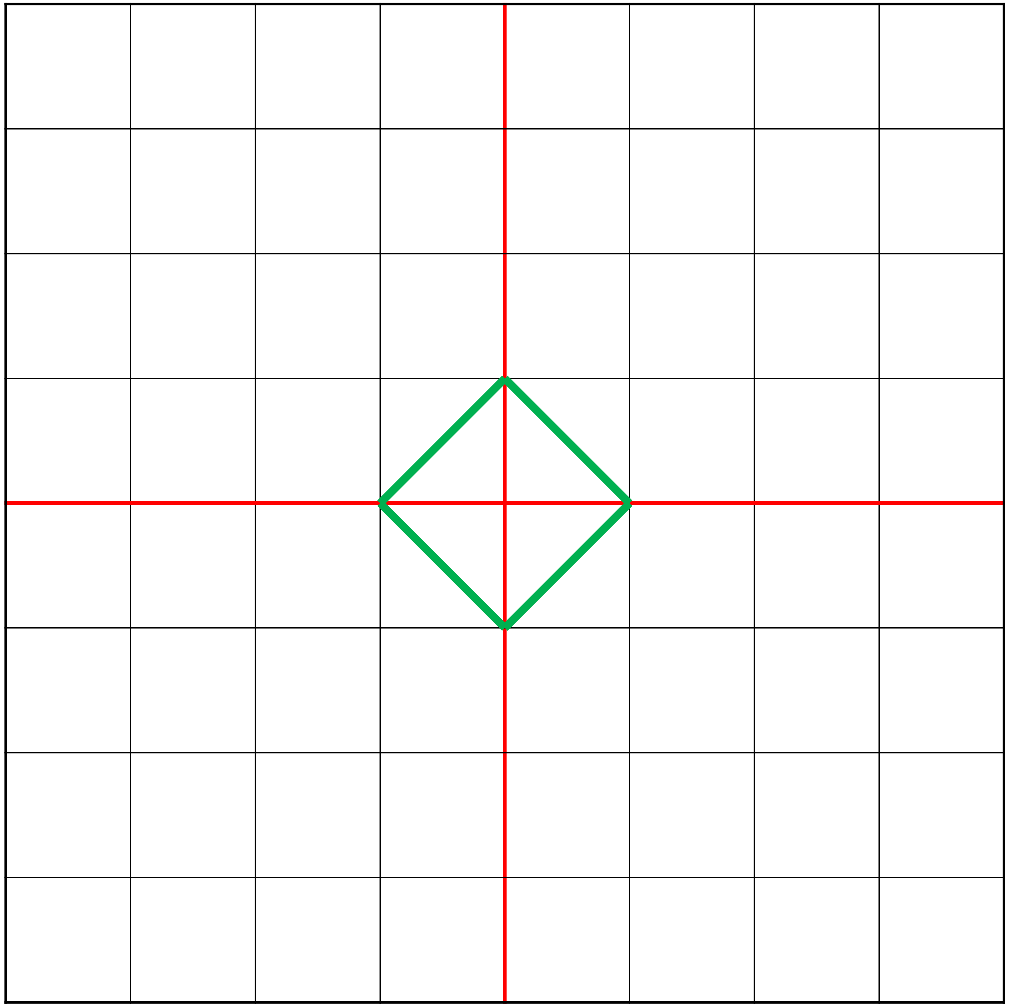
- 原点から遠ざかる方向とは？
  - 移動した向きが分からない命令は動かないことにしたときのドローンの最終的な位置を (x, y) とする
  - $x \geqq 0$ ならば **?** を **R** にすることで遠ざかることが、$x < 0$ ならば **?** を **L** にすることで遠ざかることができる
  - **?**を同じ向きにのみ移動するように置換するのが最適

距離の最小値(101 点)

- 最終的な位置が原点に近づくように移動したい

- 原点に近づく方向とは？
  - 移動した向きが分からない命令は動かないことにしたときのドローンの最終的な位置を (x, y) とする
  - $x > 0$ ならば **?** を **L** にすることで近づくことが、
    $x < 0$ ならば **?** を **R** にすることで近づくことができる
  - $y > 0$ ならば ? を **D** にすることで近づくことが、
    $y < 0$ ならば ? を **U** にすることで近づくことができる
  - $(x, y) = (0, 0)$ のときは原点から遠ざかるしかできない！

コーナーケース例: ?



原点から移動して原点に近づくことはできない！

解法

- 移動した向きが分からない命令は動かないことにしたときのドローンの最終的な位置を (x, y) とする
- その後、移動した向きが分からなかった命令を以下のように貪欲にシミュレーションする
  - 最大値を求める場合は、原点から遠ざかる方向に移動する命令と考えてシミュレーション
  - 最小値を求める場合は、原点に近づく方向に移動する命令と考えてシミュレーション
    - 原点にいる場合は遠ざかるしかできないことに注意
- 計算量は $O(|S|)$

別解

- 移動した向きが分からない命令は動かないことにしたときのドローンの最終的な位置を (x, y) とする
- 移動した向きが分からなかった命令の数を k とする

- 距離の最大値は $|x| + |y| + k$、距離の最小値は $\max(|S| \,\%\, 2, |x| + |y| - k)$ で表せる

- こちらも計算量は$O(|S|)$

C: オセロ

問題概要

- N 個のオセロの駒を黒の面が上向きになるように一列に並べる
- その後、ある区間にある駒を全て裏返すという操作を Q 回行う
- 最終的な盤面を求めよ

部分点(60点)

- $N, Q \leqq 2{,}000$ と小さい
- 愚直にひっくり返すのをシミュレーションしよう！

- 一回の操作につき最大で N 個の駒を裏返す
- 計算量は$O(NQ)$

満点(100点) 考察

- 石の数も操作回数も多いので愚直にシミュレーションしては間に合わない

- まとめてシミュレーションのようなことは可能か？

- ここで
  - 偶数回裏返された駒は黒の面が上を向く
  - 奇数回裏返された駒は白の面が上を向く

  ことに着目しよう

- 各駒について裏返された回数だけ分かればよい

満点(100点) 考察

- ある範囲に裏返した回数を 1 加算する
  という操作ができれば解けることがわかった

- これを実現する方法はいくつか存在する
  - セグメント木
  - BIT
  - etc.
- ここでは**いもす法**と呼ばれるテクニックを紹介する

いもす法

- いもす法は加算、構築、取得の 3 つの処理からなる
  - 加算: 区間$[l, r]$ に v だけ加算する
  - 構築: ある位置の値が求められるようにする
  - 取得: ある位置の値を求める

- 加算 $O(1)$, 構築 $O(N)$, 取得 $O(1)$ を実現する
  - ただし、構築後にさらに加算する、構築せずに取得するなどの操作は許されない

具体的な処理

- 基本的なアイデアは以下の 2 つ
  - 差分を覚えておく
  - 累積和により構築する

- 加算処理では
  - l 番目の値に v 加算する
  - $r+1$ 番目の値に $-v$ 加算する
- 構築処理では
  - i $(1 \leqq i \leqq N-1)$番目の値を $i+1$ 番目の値に加算する
- すると各位置の値がそれぞれ求まる

愚直シミュレーション

いもす法

1	2	3	4	5	6	7
0	0	0	0	0	0	0

- 初期状態

1	2	3	4	5	6	7
0	0	0	0	0	0	0

1	2	3	4	5	6	7
0	2	2	2	0	0	0

- [2, 4] に 2 を加算

1	2	3	4	5	6	7
0	2	0	0	-2	0	0

1	2	3	4	5	6	7
0	2	2	5	0	0	0

- [4, 4] に 3 を加算

1	2	3	4	5	6	7
0	2	0	3	-5	0	0

1	2	3	4	5	6	7
1	3	3	6	1	1	0

- [1, 6] に 1 を加算

1	2	3	4	5	6	7
1	2	0	3	-5	0	-1

1	2	3	4	5	6	7
1	3	3	6	1	0	0

- いもす法の方は
累積和を取る

1	2	3	4	5	6	7
1	3	3	6	1	1	0

1	2	3	4	5	6	7
1	3	3	6	1	0	0

- 最終的な結果は同じ！！

1	2	3	4	5	6	7
1	3	3	6	1	0	0

解法

- 各操作を「区間 $[l, r]$ に 1 加算する」処理とみなす

- いもす法など適当な方法によりこれを処理する

- それぞれの駒について、偶数回裏返されたならば 0 、奇数回裏返されたならば 1 を出力する

- 計算量は $O(N+Q)$

D: トレジャーハント

問題概要

- N 頂点、M 本の重みつき有向辺からなる有向グラフが与えられる
- 頂点 i に 1 分間滞在すると所持金が A_i 円増える
- 時刻 0 に頂点 1 から出発し、 T 分後に頂点 1 に戻る
- 所持金を最大化せよ

考察

- 滞在する必要がある頂点は 1 つと考えてよい
  - 複数の頂点に滞在することにより、所持金の最大値がより大きくなるということはありえない

- この問題は、「ある頂点 i に出来るかぎり早く到着し、出来るかぎり滞在してから、頂点 1 に出来るかぎり早く帰る」という問題だと分かった

考察

- 滞在する必要がある頂点は 1 つと考えてよい
  - 複数の頂点に滞在することにより、所持金の最大値がより大きくなるということはありえない

- この問題は、「ある頂点 i に**出来るかぎり早く到着**し、出来るかぎり滞在してから、頂点 1 に**出来るかぎり早く帰る**」という問題だと分かった

- これは**最短経路問題**と呼ばれる有名問題

部分点(50点)

- 以下の 2 種類の最短距離を全て求めよう！
  1. 頂点 1 から、頂点 i への最短距離
  2. 頂点 i から、頂点 1 への最短距離

- ワーシャルフロイド法を使うと $O(N^3)$
- ダイクストラ法を使うと $O(NM\log N)$

満点(100点) 考察

- 部分点と同様、以下の 2 種類の最短距離を求めたい
    1. 頂点 1 から、頂点 i への最短距離
    2. 頂点 i から、頂点 1 への最短距離

- 種類 1 はダイクストラ法により高速に求められる
- 種類 2 で全ての頂点から調べては間に合わない

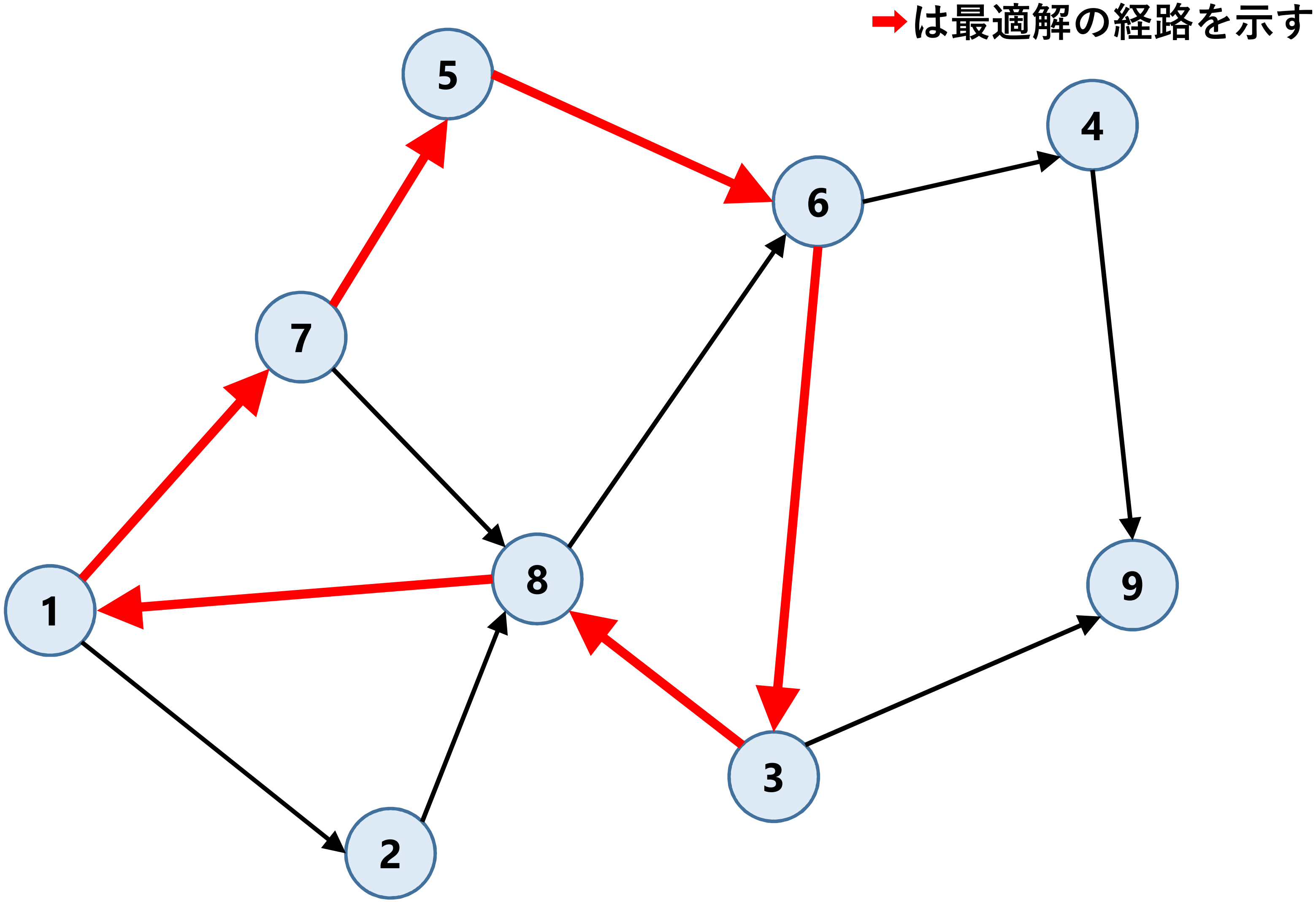
満点(100点) 考察

- 部分点と同様、以下の 2 種類の最短距離を求めたい
    1. 頂点 1 から、頂点 i への最短距離
    2. 頂点 i から、頂点 1 への最短距離

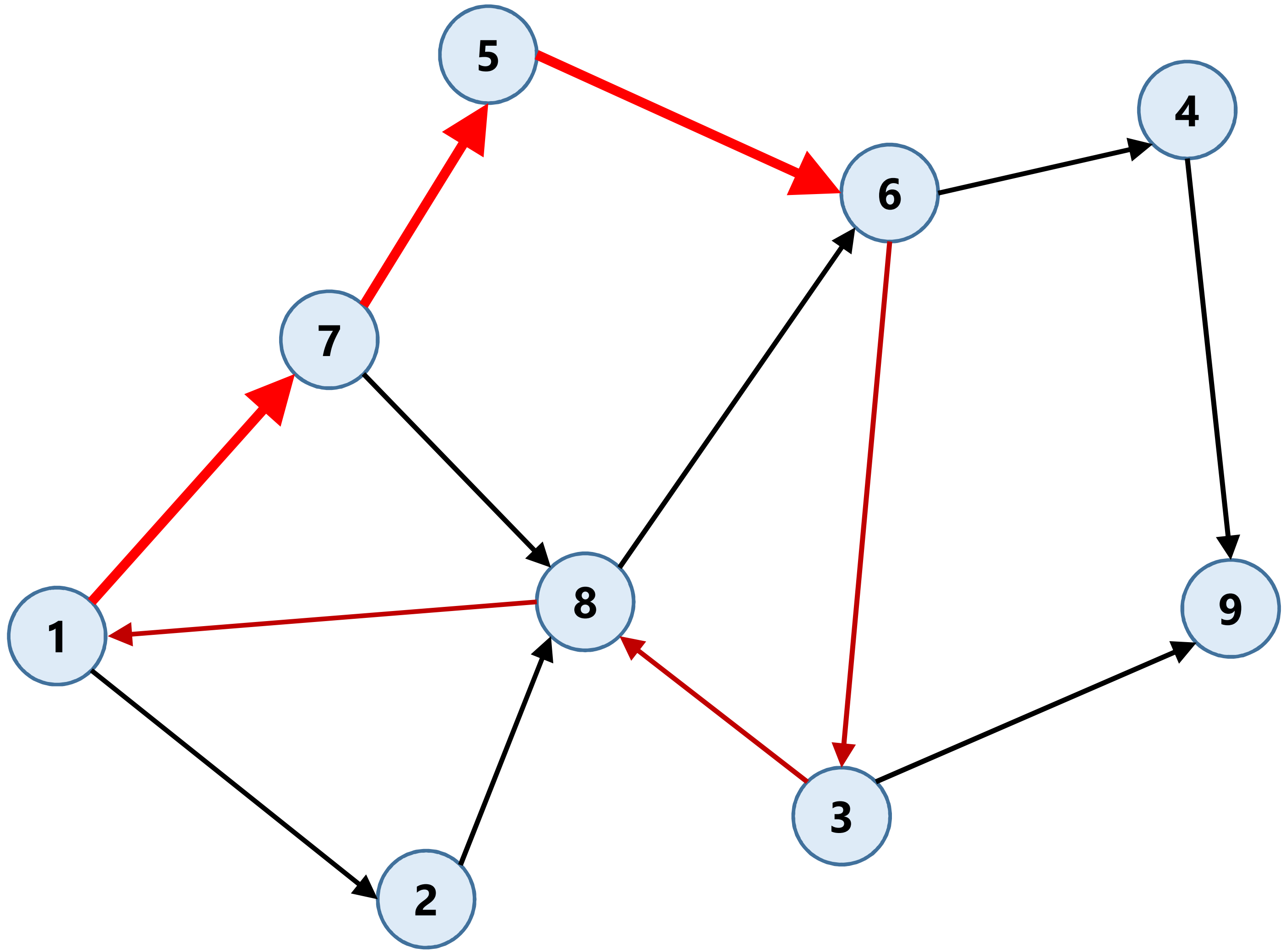
- 種類 1 はダイクストラ法により高速に求められる
- 種類 2 で全ての頂点から調べては間に合わない

- 頂点 1 から辺を**逆向きに辿る**ことを考える

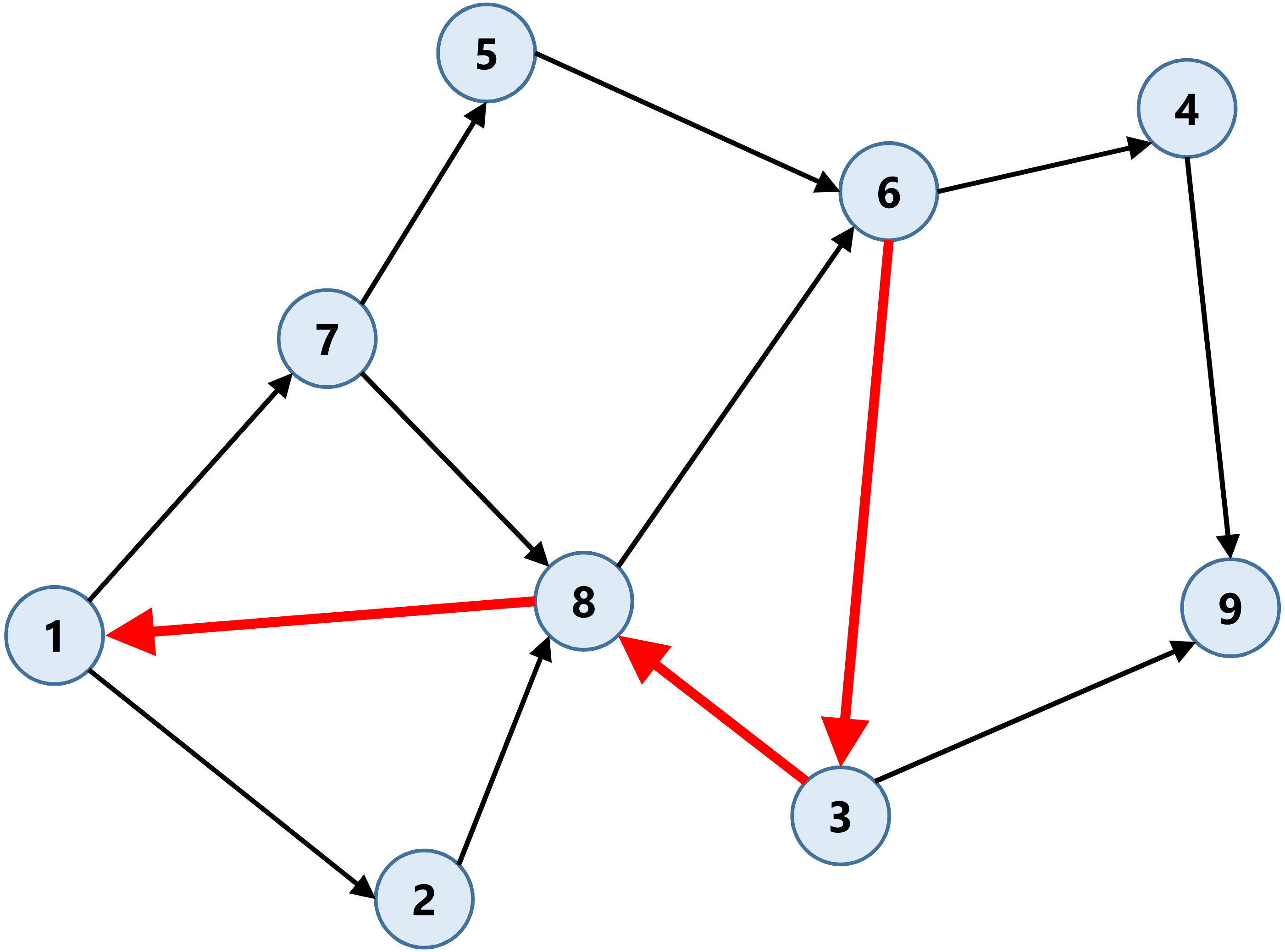
頂点6に滞在するのが最適解のとき



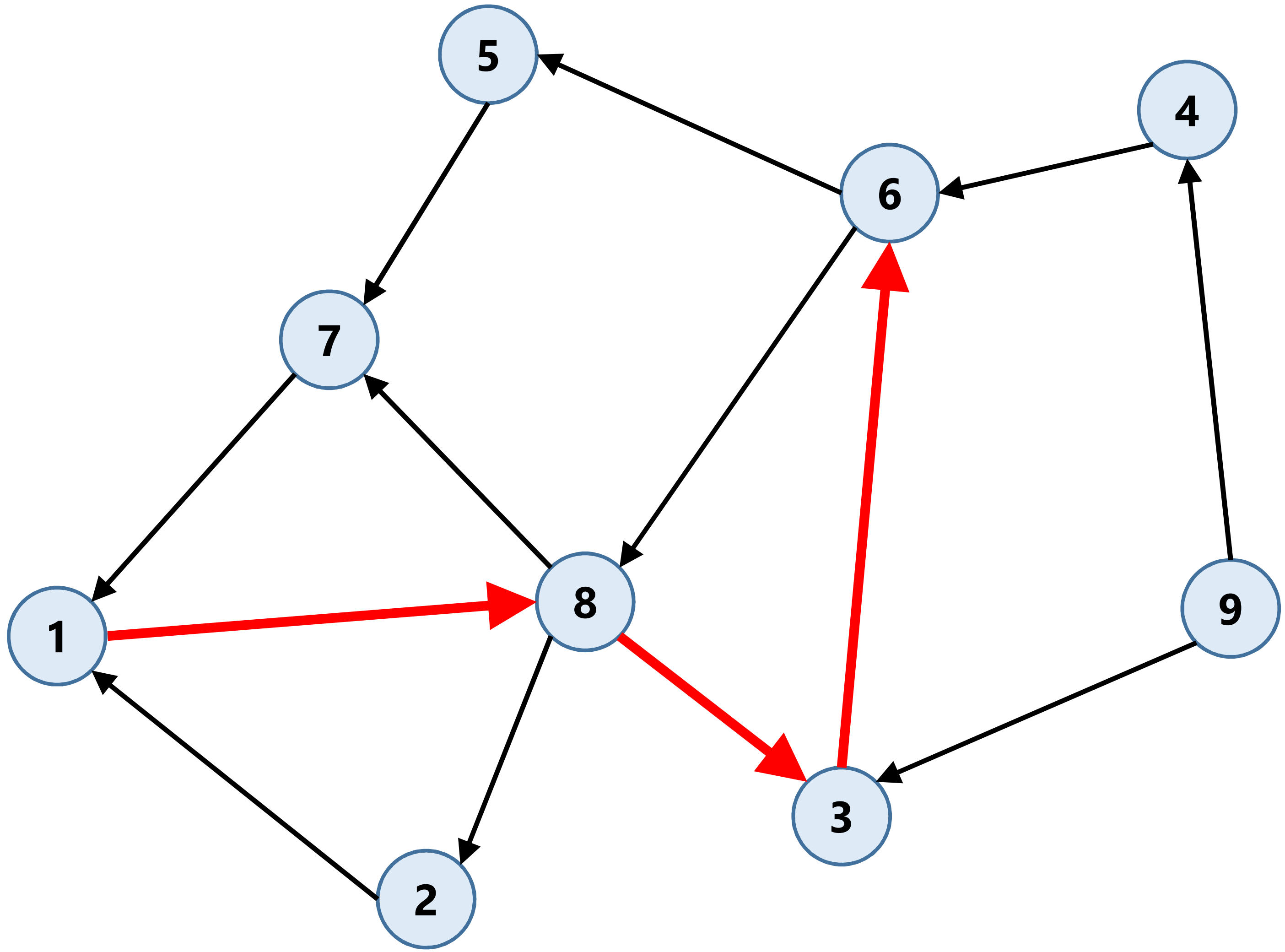
行く方向



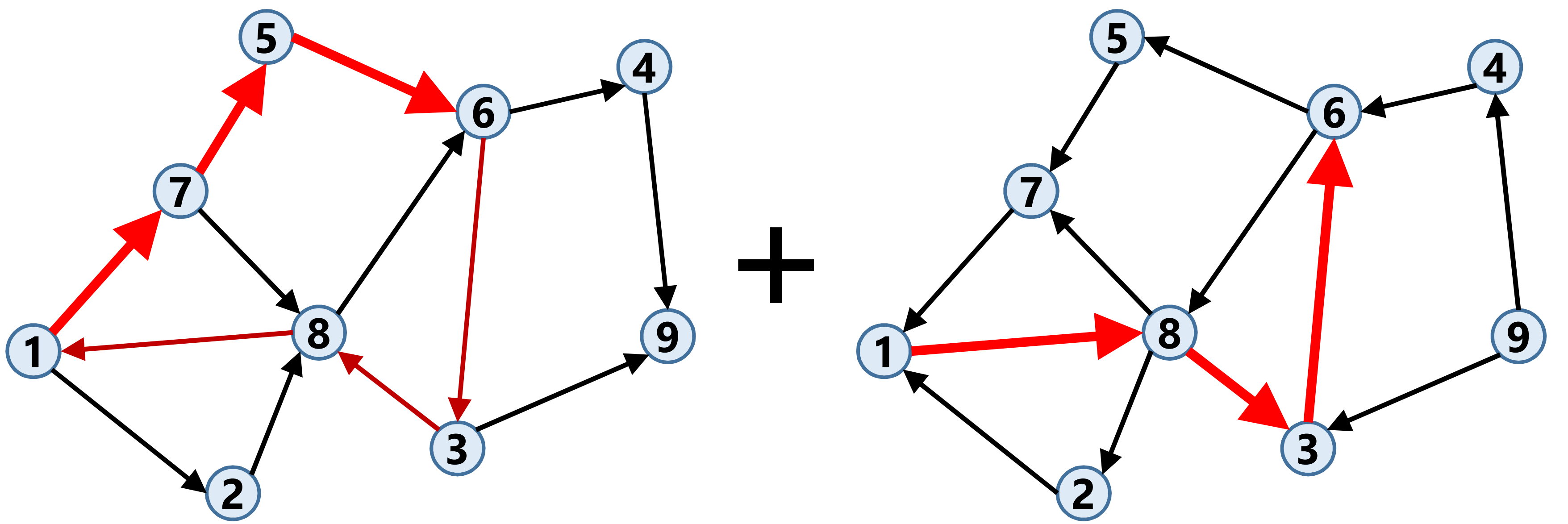
帰る方向



辺の向きをひっくり返す



二つの単一始点最短経路問題！



考察

- 結局、頂点 i から頂点 1 への最短距離というのは
    - 頂点 1 から辺を逆向きに辿ったときの頂点 i への最短距離

  と言い換えられる

- 最短距離を求めるのは行きと帰りの 2 回だけで十分

解法

- 各頂点について、頂点 1 からの最短距離を辺の向きがそのままの場合と、辺の向きを逆転した場合について求める

- ここから各頂点に滞在可能な最大の時間を求めて、所持金の最大値を全探索する

- 計算量は$O((N+M)\log N)$

注意

- 答えは 32 ビット整数に収まらない場合があります
  - $T = 10^9$, $A_1 = 10^5$ など