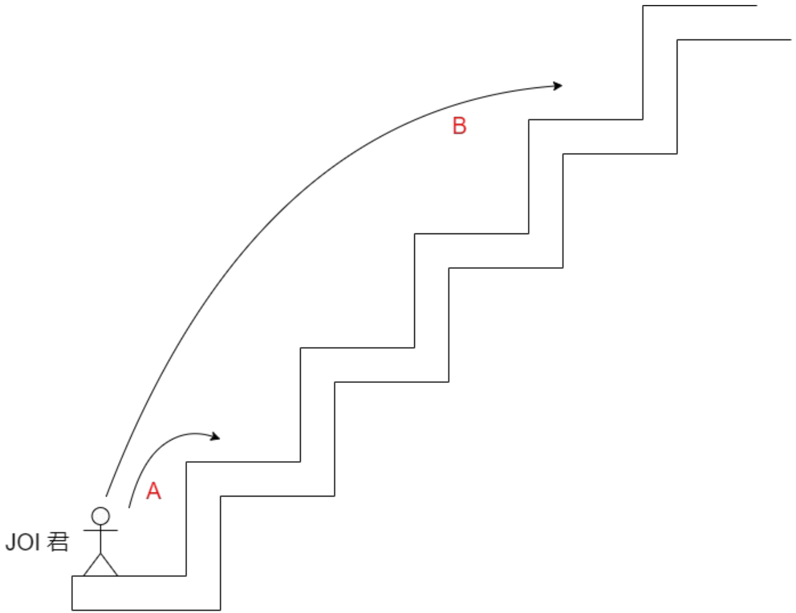
JOI 2023/2024 春季トレーニング Day 3
塔 (Tower) 解説

解説担当: 菅井

問題概要

- 階段を上る
- 「A 秒かけて 1 段上る」or「B 秒かけて D 段上る」ができる
- N 個の区間で工事が行われており、入ることができない
- Q クエリ: X_j 段目にたどり着けるか？たどり着けるなら何秒かかるか？
- 制約: (工事の個数) <= 2*10^5, (クエリの個数) <= 2*10^5, (段数) <= 10^12



小課題

小課題 1 (5 点) 段数が 10^6 以下

小課題 2 (38 点) N, Q <= 2000

小課題 3 (25 点) A=1, B=D

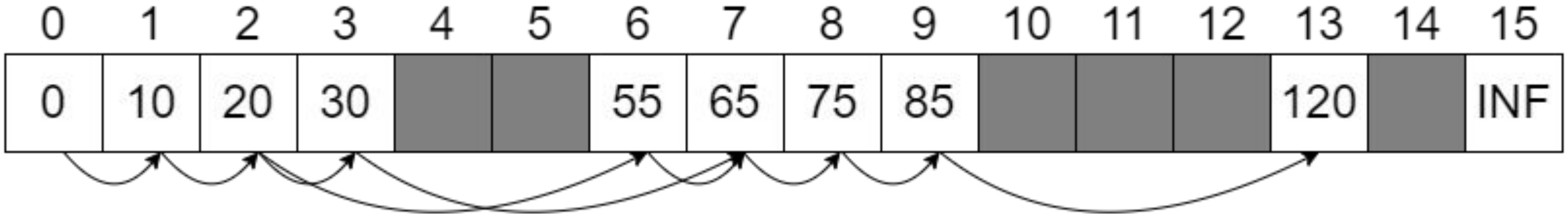
小課題 4 (32 点) 追加の制約はない

小課題 1 (5 点)

- 段数が 10^6 以下
- DP ができる
- dp[i] = (0 段目から i 段目まで行くのにかかる最短時間 (不可能なら INF))
- 5 点獲得

小課題 2 (N, Q <= 2000)

サンプル 1 の DP を眺めよう



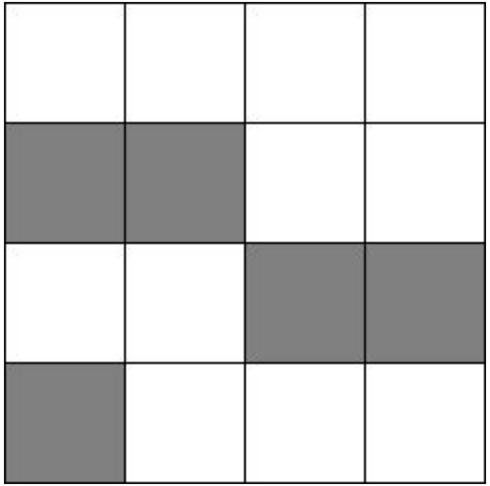
この問題の大きな特徴:「D 段上に上る」

D が定数 → mod D の値が重要になりそう

小課題 2 (N, Q <= 2000)

D で割った余りと商に注目

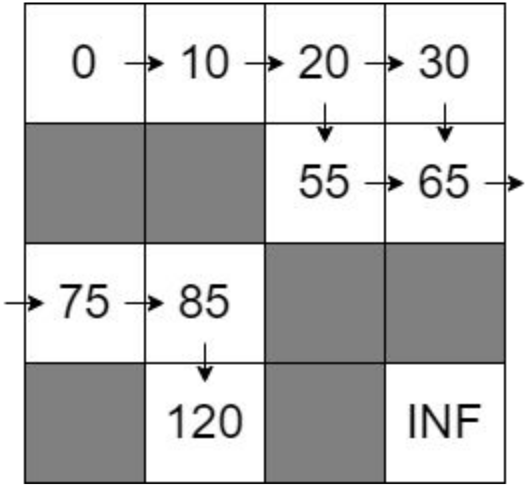
マスを横 D 列のグリッド上に配置できる



小課題 2 (N, Q <= 2000)

グリッド上でできる移動

- 右に 1 進む
- 下に 1 進む
- 右端から次の行の左端へワープする

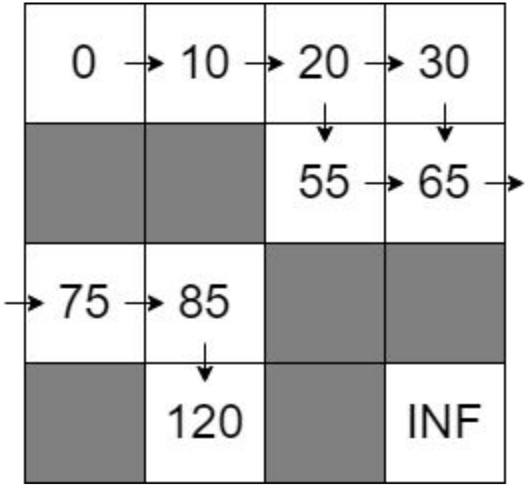


小課題 2 (N, Q <= 2000)

グリッドに上に障害物とクエリを配置して、DP を行いたい

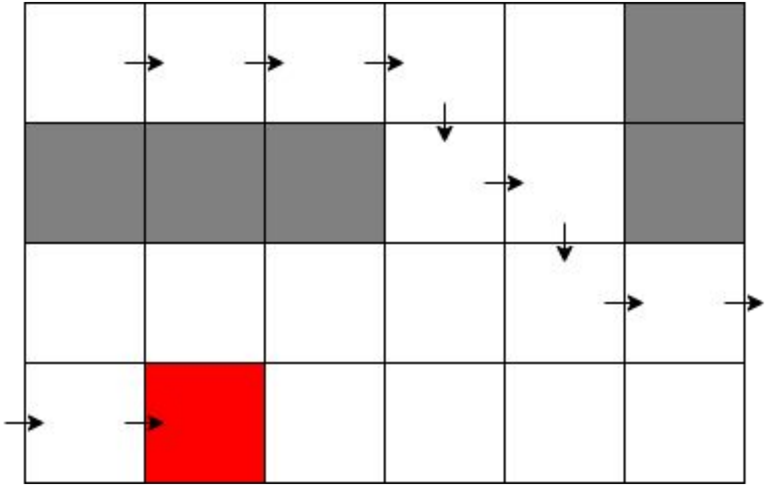
座標圧縮を行うと、縦横 O(N+Q) マスにできる

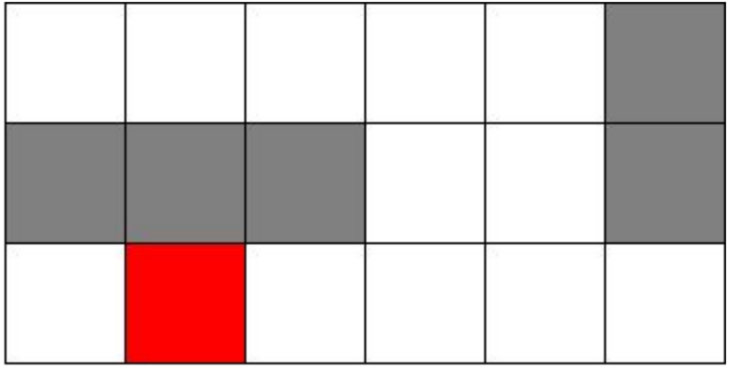
→ 全体で O((N+Q)^2)



小課題 2 (N, Q <= 2000)

何もない行を適当に消すと答えがずれるので、注意しましょう





小課題 3 (A=1, B=D)

どちらの移動方法でも、1 秒あたり 1 段上っている

→ X_j 段目にたどり着けるなら、どのようにしても X_j 秒かかる

→ X_j 段目にたどり着けるかどうかだけ判定できればよい！

小課題 3 (A=1, B=D)

小課題 2 の座標圧縮を行った後、DP を行ごとに見ていく

i-1 行目の DP から、i 行目の DP はどのように得られるか？

0	0	1	0	0	1	0	1	1	1
?	?	?	?	?	?	?	?	?	?

小課題 3 (A=1, B=D)

① とりあえずそのままコピーする

0	0	1	0	0	1	0	1	1	1
0	0	1	0	0	1	0	1	1	1

小課題 3 (A=1, B=D)

② 障害物があるところは 0 にする

0	0	1	0	0	1	0	1	1	1
0	0	1	0	0	1	0	0	1	1

小課題 3 (A=1, B=D)

③ i-1 行目の右端のマスが 1 なら、左端のマスを 1 にする

0	0	1	0	0	1	0	1	1	1
1	0	1	0	0	1	0	0	1	1

小課題 3 (A=1, B=D)

④ 障害物がない区間を列挙し、最も左にある 1 を見つけ、それ以降をすべて 1 にする

0	0	1	0	0	1	0	1	1	1
1	1	1	0	0	1	1	0	1	1

小課題 3 (A=1, B=D)

① とりあえずそのままコピーする

これは、i-1 行目の DP 配列を使いまわせばよい

小課題 3 (A=1, B=D)

② 障害物があるところは 0 にする

③ i-1 行目の右端のマスが 1 なら、左端のマスを 1 にする

④ 障害物がない区間を列挙し、最も左にある 1 を見つけ、それ以降をすべて 1 にする

これらは、遅延セグメント木で処理できる (区間の最も左の 1 の位置を持つ)

満点

DP の遷移は以下のようになる:

① 全体に定数を加算

② 障害物があるところは INF にする

③ chmin(i 行目の左端, i-1 行目の右端 + A)

④ 障害物がない区間を列挙し、順に chmin(dp[j + 1], dp[j] + A)

満点

定数だけずらして、以下のようにできる:

① 障害物があるところは INF にする

② chmin(i 行目の左端, i-1 行目の右端 + x)

③ 障害物がない区間を列挙し、順に chmin(dp[j + 1], dp[j])

満点

実は、各行の DP 配列は、必ずある整数 k を用いて

((k+1)x と INF からなる列) + (kx と INF からなる列) と表せる

証明: 帰納法

前半と後半の境界を考えることができる

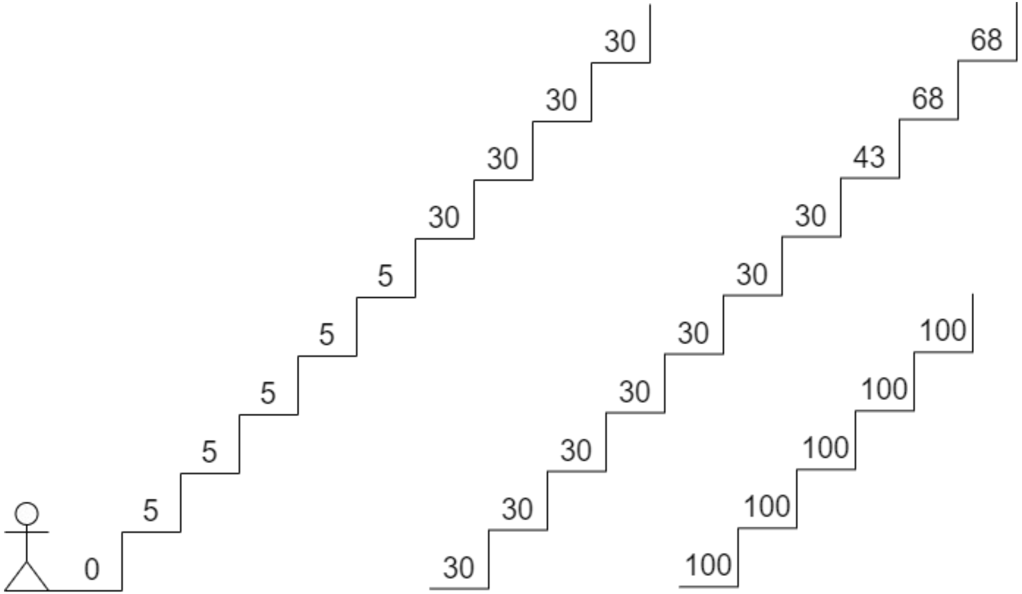
満点

「各要素が INF かどうか」は、小課題 3 と同様の遅延セグメント木で管理できる

DP を更新するとき、前半と後半の境界を適切に更新すればよい

計算量は $O((N+Q)\log(N+Q))$ など

得点分布



0	5	5	5	5
5	30	30	30	30
30	30	30	30	30
30	30	43	68	68
100	100	100	100	100