Teleporter 解説

解説担当者：blackyuki

Subtask 0

問題概要

0 問題概要

問題概要

- 半開区間 [S_1, T_1), [S_2, T_2), …, [S_M, T_M) がある。
- 区間を破壊することで、区間スケジューリングの答えが K 以下となるようにしたい。
- 区間 i はコスト C_i で破壊できる。
- コストの総和の最小値を求めてください。

0 問題概要

制約

- N, M <= 10^5
- 1 <= S_i < T_i <= N
- 1 <= C_i <= 10^9

Subtask 1

$K=1$

1 K=1

- 「区間スケジューリングの答えが 1 」を同値に言い換える。
  - → max S_i < min T_i
- j = 1, 2, …, N-1 について、S_i <= j < T_i となる i の C_i の総和を計算すれば良い。
- 累積和で全体 O(N+M)。
- 5点がもらえます。

1 K=1

- （別解）：双対をとる
  - 「区間スケジューリングの答えが K」は次と同値
  - 「全ての区間を貫通させるには K 本の串が必要」
  - K=1 の場合、先ほどの解法に一致する

Subtask 2

N,M <= 20

2 N,M <= 20

- bit 全探索をします。
- 3 点がもらえます。
- うれしいね。

Subtask 3

N,M <= 500

3 N,M <= 500

- 区間スケジューリング問題の貪欲アルゴリズムを思い出す
  - T_i の昇順（タイブレークは S_i の昇順）で区間をソートし貪欲に選んでいく
- 便宜上、次の2つの区間を番兵として追加する
  - (S_0, T_0, C_0) = (-1,0,inf)
  - (S_{M+1}, T_{M+1}, C_{M+1}) = (N, N+1, inf)

3 N,M <= 500

- 貪欲に取っていった結果区間 0=i_1< i_2< …< i_k=M+1 が選ばれるための条件を考える
- 区間 i_1 を選んでから区間 i_2 を選ぶまでに、本来選ぶべきだった区間は全て削除する必要がある
    - 具体的には、i_1 < j < i_2 かつ S_j >= T_{i_1} を満たす区間 j

3 N,M <= 500

- 次のような dp を考える
  - dp[k][i] : 区間 i まで見て、区間 i を含む k 個の区間を選んだ場合のコストの総和の最小値
  - dp[k][i] -> dp[k+1][j] の遷移コストは２次元累積和で O(1) 取得可能
- 計算量は O(KM^2)

3 N,M <= 500

- (別解)：双対をとる。
  - 番兵として串 0 と串 N+1 は必ず刺すことにする。
  - 串 $0 = i_1 < i_2 < \ldots < i_k = N+1$ を選んだ時、削除するべき区間のコストの最小値を求めたい。
  - i_1 から i_2 に遷移する時に加算する必要のあるコスト：
    - $i_1 < S_j < T_j <= i_2$ を満たす区間 j のコストの総和

3 N,M <= 500

- (別解)：双対をとる。
  - 本解と似たような dp が組める。
  - 二次元累積和を用いて O(1) 遷移が可能。
  - 計算量は O(KN^2)。

Subtask 4

N,M <= 4000

4 N,M <= 4000

- 以下、双対をとったバージョンで説明します。
  - 双対を取らない場合もそんなに変わらないはずです。
- DP を高速化したいです。
- dp[k][i_1] -> dp[k][i_2] の遷移コストを F[i_1][i_2] とします。
  - F[i_1][i_2] = (i_1 < S_j < T_j <= i_2 を満たす区間 j のコストの総和)
- このコスト関数 F は Monge です。

4 N,M <= 4000

- Monge とは …？

Monge の定義

実数からなる $N \times M$ 行列について考えます。

> $N \times M$ 行列 A が Monge であるとは、以下の条件が成り立つことをいう。
>
> - $1 \le i_1 < i_2 \le N$ および $1 \le j_1 < j_2 \le M$ を満たすすべての整数の組 (i_1, i_2, j_1, j_2) について、$A_{i_1,j_1} + A_{i_2,j_2} \le A_{i_1,j_2} + A_{i_2,j_1}$ である。

引用：https://hackmd.io/@tatyam-prime/monge1

4 N,M <= 4000

- 証明
  - 一つの区間が F に与える寄与が Monge なので、

    それらの和をとっても Monge になります。

4 N,M <= 4000

- Monge 性の利用による DP 高速化

小課題3

- Monotone Minima なので分割統治で O(NlogN)
- 全体 O(N^2 log N) で定数は軽い
- 分からない場合は「Monge DP 高速化」等で検索してください

引用：https://www2.ioi-jp.org/camp/2023/2023-sp-tasks/contest3/chorus-review.pdf

4 N,M <= 4000

- Monge 性の利用による DP 高速化
  - dp[k+1][r] の最小値が dp[k][l] 由来であることを f(k,r)=l と書くことにすると、f(k,1)<=f(k,2)<=… となる。
  - monotone minima ができる。
  - 計算量は O(KNlogN)

4 N,M <= 4000

- (別解) Monge 性を使わずに、遅延セグ木を用いた inline DP をやっても良い
  - 定数倍はだいぶ厳しい (monotone minima の定数倍はすごく軽い)
  - 頑張る必要がある
  - 遅延セグ木を使わない解法も
    - 詳細は subtask 6 で

Subtask 5,6

$N, M <= 40000$

$N, M <= 100000$

5 N,M <= 40000 (sub5), 100000 (sub6)

- Alien DP

- N 頂点のグラフがあり、頂点 i から頂点 j へ $(i < j)$ に移動するコストが $A_{i,j}$ であるとします。A が Monge であるとき、単一始点最短路が $O(N)$ で求められます。
  - 辺をちょうど d 回通る場合の $s-t$ 最短路が $O(N \log A_{\max})$ で求められます。

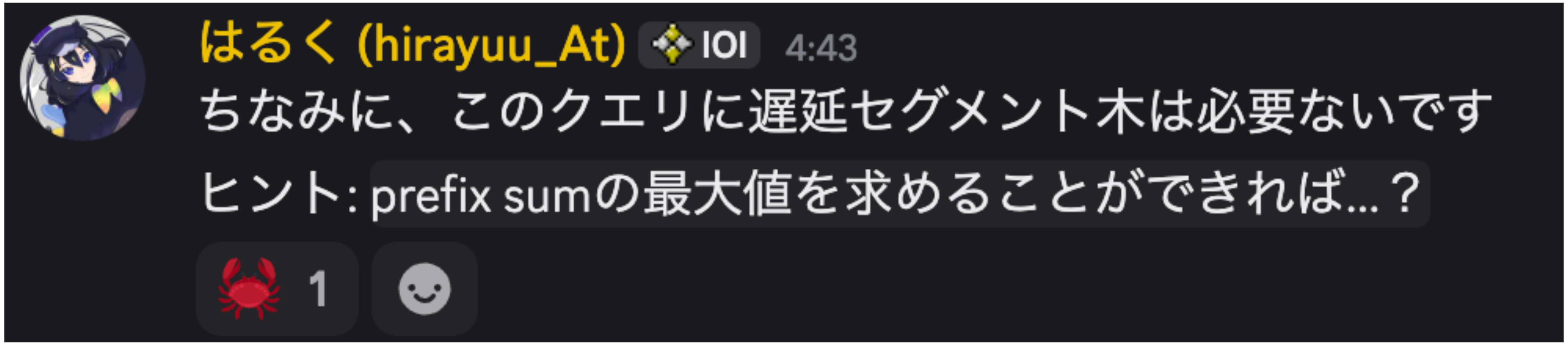
引用：https://hackmd.io/@tatyam-prime/monge1

5　N,M <= 40000 (sub5), 100000 (sub6)

- Alien DP
  - 各辺のコストに X 加算した時の最短距離が
    高速に求められれば良い
  - 選択肢1：区間加算・区間 max 取得の遅延セグ木
  - 選択肢2：starry sky tree
  - 選択肢3：std::set で隣接点の差分を管理＋不要な点を削除
    - スライド最小値のようなイメージ

5 N,M <= 40000 (sub5), 100000 (sub6)

- Alien DP
  - 選択肢 4：普通のセグ木で差分を管理し prefix sum の max



  - 昨日の問題をちゃんと復習しましたか？という問題でしたね

5 N,M <= 40000 (sub5), 100000 (sub6)

- 計算量は O(NlogNlog(NmaxC))
  - 定数倍が良ければ満点が得られる
  - 余計な log がついたり定数倍が悪ければ sub5 が得られる

得点分布

6 得点分布

- 84点：1人
- 60点：3人
- 37点：7人
- 32点：5人
- 8点：7人
- 3点：4人
- 0点：3人