総和占い
Fortune Telling 2

今西 健介(@japlj)

JOI Open Contest 2014, Day1

問題概要

問題

- 両面に整数 A_i, B_i が書かれたカードが N 枚ある
  - 最初は A_i が書かれた面が見えている
- 「見えている整数が T_j 以下のカードを裏返す」という操作を K 回行う
- すべての操作後に見えている整数の合計値は？

満点制約

- $1 \leqq N, K \leqq 200{,}000$

小課題 1（4点）

制約

- $1 \leqq N \leqq 1{,}000$
- $1 \leqq K \leqq 1{,}000$

解法

- 各操作をそのまま実装する
  - 1 回の操作の計算量は $O(N)$
  - 合計で $O(NK)$

小課題 2（31点）

制約

- $1 \leqq N \leqq 40{,}000$
- $1 \leqq K \leqq 40{,}000$

解法

当然、小課題 1 のままでは解けないので

何らかの工夫が必要

→操作について詳しく考えてみよう

考察、その前に

仮定

- ここから先では全てのカードで $A_i \leqq B_i$ と仮定
  - 説明のしやすさ、分かりやすさのため

仮定？

- じゃあ $B_i < A_i$ なカードはどうなるの？
- A_i, B_i を交換し、最初 B_i の面が見えていると思う
  - 面倒そうな方法に聞こえるけど、これから説明する解法ではこう考えておいたほうが楽

「操作」は何をしているか

操作

- 見えている整数が T_j 以下のカードを裏返す

カード視点で見る

- $T_j < A_i$ のとき → 必ず裏返されない
- $A_i \leqq T_j < B_i$ のとき → A_i の面が上なら裏返される
- $B_i \leqq T_j$ のとき → 必ず裏返される

A_i, B_i と T_j の大小関係によって操作の内容が決まる

大小関係による分類

カードの同一視

- (A_i, B_i) がそれぞれ $(3, 6), (2, 5)$ の 2 枚のカード
- それらに対する 3 回の操作 $T = (4, 9, 1)$
- 両方のカードに対し、操作の内容は全て同じ！

操作	カード	カード	
4	$3 \leqq 4 < 6$	$2 \leqq 4 < 5$	→ A_i が上なら裏返す
9	$6 \leqq 9$	$5 \leqq 9$	→ 裏返す
1	$1 < 3$	$1 < 2$	→ 裏返さない

大小関係による分類

カードの同一視

- 同一視できるカードたちについては
  それぞれに対して操作の内容を考える必要がない
  - そのうち 1 枚に対して考えれば、後は同じ
- カードを分類することで計算量が削減できる？

カードの種類数

- T は K 個の値からなる
- 大小関係で分類してもまだ $O(K^2)$ 通りもある

分類してうまくいく場合

カードの種類数を抑える

- K (操作の回数) が小さければ、種類数も小さい
- 操作をいくつかの区間に分けて考えよう！
  - 各区間の中では操作の回数は少ない
  - その区間の操作たちをカードの分類によって高速処理

具体的に

- 1区間に B 個の操作を行うとする
  - B の値をうまく決めると……？

バケットサイズ

計算量

- 各区間には B 個の操作がある
  - カードを $\mathrm{O}(B^2)$ 種類に分類できる
  - カードの処理はまとめて $\mathrm{O}(B^3)$ でできる
  - よって、各区間の計算量は $\mathrm{O}(N+B^3)$
- 区間の個数は K/B 個
- 全体の計算量は $\mathrm{O}(NK/B+B^2K)$ となる
  - $B=K^{1/3}$ とおけば $\mathrm{O}(NK^{2/3}+K^{5/3})$

分類処理や
実際に裏返す処理などに
$\mathrm{O}(N)$ かかることに注意

これで小課題 2 が解ける！

小課題 3（65点）

制約

- $1 \leqq N \leqq 200{,}000$
- $1 \leqq K \leqq 200{,}000$

解法

操作についてさらに考察を深める！

「操作」は何をしているか(再)

操作

- 見えている整数が T_j 以下のカードを裏返す

カード視点で見る

- $T_j < A_i$ のとき → 必ず裏返されない
- $A_i \leqq T_j < B_i$ のとき → A_i の面が上なら裏返される
- $B_i \leqq T_j$ のとき → 必ず裏返される

A_i, B_i と T_j の大小関係によって操作の内容が決まる

「操作」は何をしているか

操作

- 見えている整数が T_j 以下のカードを裏返す

カード視点で見る

これは考えなくていい

- $T_j < A_i$ のとき → 必ず裏返されない
- $A_i \leqq T_j < B_i$ のとき → A_i の面が上なら裏返される
- $B_i \leqq T_j$ のとき → 必ず裏返される

とは一体……？

何も考えずに裏返せばよい

「操作」は何をしているか

操作

- 見えている整数が T_j 以下のカードを裏返す

カード視点で見る

これは考えなくていい

- $T_j < A_i$ のとき → 必ず裏返されない
- $A_i \leqq T_j < B_i$ のとき → **操作後、B_i の面が上を向く**
- $B_i \leqq T_j$ のとき → 必ず裏返される

こういうことだ！

何も考えずに裏返せばよい

操作の性質

重要な事実

- $A_i \leqq T_j < B_i$ なる操作 T_j を行ったあと、
  上を向く面は以前の状態によらず B_i の面になる
  - これは**「大きい方を上に向ける操作」**と言える
- 大きい方を上に向ける操作を行うと、
  それ以前に行われた操作のことは忘れてもよい！
  - この操作によってカードの状態がリセットされる

1 枚のカードから見た操作列

操作の分類

- カードから見れば操作は 3 種類に分類できる
  - 何もしない(P)、裏返す(Q)、大きい方を上に向ける(M)

- 例として以下の様な操作列を考える

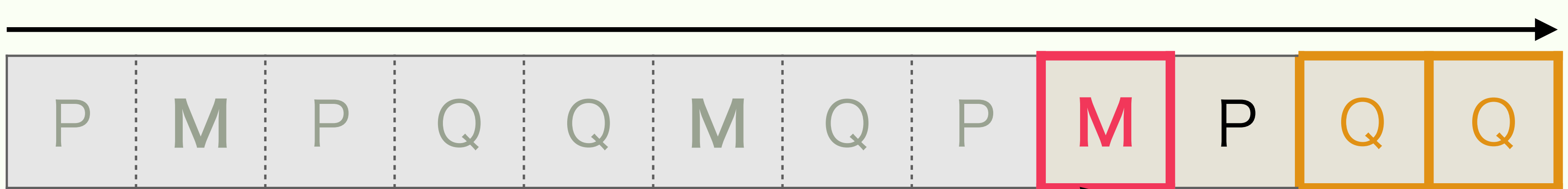
P	M	P	Q	Q	M	Q	P	M	P	Q	Q

1 枚のカードから見た操作列

操作の分類

- カードから見れば操作は 3 種類に分類できる
  - 何もしない(P)、裏返す(Q)、大きい方を上に向ける(M)

- 例として以下の様な操作列を考える



実質行う操作は
これらだけ

ここの操作を行った時点で
それ以前の操作はなかったことにしてよい

解法アウトライン

解法の方針

- 各カードごとに以下のように処理する:
- $A_i \leqq T_j < B_i$ なる最も後ろの T_j を探す
- 見つけた T_j 以降で $B_i \leqq T_k$ なる k の個数を求める
- これによりカード i の最終的な向きがわかる
  - 大きい方を上に向ける操作がないときに注意

最後の
「大きい方を上にする」操作

実装

- そのまま書くだけでは $O(NK)$ なので、工夫が必要

実装

実装方針

- 色々な方法がありますが、たとえば
  - T_j を座標圧縮し、位置 T_j に値 j を書いておく
  - A_i, B_i に挟まれる区間の中の最大値をとってくる
- といった感じで
  - segtree, BIT, RMQ 等の応用的な使い方

細かい実装の差にもよるが
$O((N+K) \log (N+K))$ や
$O((N+K) \log^2 (N+K))$ で満点

実装詳細

- segtree 等についての説明は省きます
  - 他に解説している資料・本がいろいろあります