L : Linear Time Inversion Number

原案:Dispersion
解説:Dispersion

問題概要

- 順列 P に対し、次の量を定める
  - Alice(P) := (P の転倒数)
  - Bob(P) := 1/2 × sum_i abs(P_i - i)
- 長さ K の数列 A が与えられる
- A を prefix にもつ、長さ N の順列 P のうち、Alice(P) = Bob(P) を満たすものを数え上げよ

- 制約
  - N ≦ 200,000

解法 (1 / 3)

- 隣接 swap によりソートする様子を考える
    - Alice(P) (転倒数) を 1 下げるとき、Bob(P) は高々 1 しか減らない
    - つまり、いかなる順列 P に対しても Alice(P) ≧ Bob(P) が成立

- Alice(P) = Bob(P) となるのはいつか？
    - P_i < i, P_i = i, P_i > i なる要素 P_i をそれぞれ、+, =, - の要素と呼ぶ
    - 隣接 swap してよいのは (+, -) の要素の組だけ

- しっかり詰めると、実は | LDS(P) | ≦ 2 と同値であると分かる！！
    - +, = の要素を取り出すと単調増加、- の要素だけ取り出しても単調増加
    - 逆向きの証明は場合分け

解法 (2 / 3)

- | LDS(P) | ≦ 2 を満たす順列 P の数え上げ
  - DP(i, j) := 整数 [0, i) を挿入した時点で、P[j : i) が昇順である (P[j-1 : i) は昇順でない) もののうち、| LDS(P) | ≦ 2 なる順列の数
  - という挿入 DP により O(N^3) or O(N^2) time で求められる

- 原案はこれを許す制約でした

解法 (2 / 3)

- | LDS(P) | ≦ 2 を満たす順列 P の数え上げ
  - DP(i, j) := 整数 [0, i) を挿入した時点で、P[j : i) が昇順である (P[j-1 : i) は昇順でない) もののうち、| LDS(P) | ≦ 2 なる順列の数
  - という挿入 DP により O(N^3) or O(N^2) time で求められる

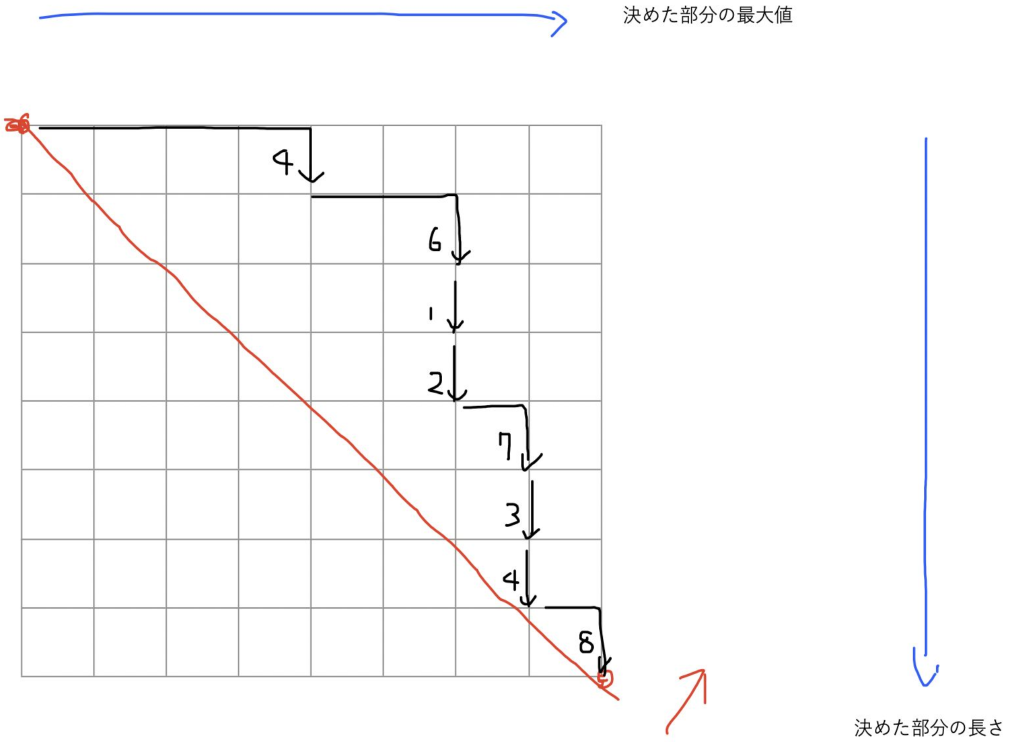
もっと改善できます

解法 (3 / 3)

- | LDS(P) | ≦ 2 なる順列とカタラン数の間には右図の全単射がある

- 先頭 K 項が固定されているので、点 (x, y) から点 (N, N) に行く経路のうち、対角線を通らないものを数え上げる
    - これは鏡像法により、二項係数の差で表せる

- 計算量は O(N + log MOD) time



統計情報

- Acceptances
    - 0 + 2 teams
- First Acceptance
    - Screenwalkers (272 min)