H: Kimagure Cleaner

原案：田中
解答：保坂、森、矢藤、古賀
解説：古賀

概要

- 平面上を動き回る掃除機がある
- 「左か右に90度回転し、ある距離だけ進む」を繰り返す
- 行動ログは残っているが、不完全である
  - 回転方向がどっちか書いていない
  - 進んだ距離が下限と上限しかわからない
- 不完全なログと最終的な位置から、完全なログを復元せよ
  - 複数あればどれでも良い

制約

- ログの行数:　1 ≦ N ≦ 50
- 進んだ距離:　1 ≦ LL ≦ LR ≦ 55,555,555
- ゴール位置:　1 ≦ X,Y ≦ 10^9

解法(1)

- N ステップ後に掃除機がいる可能性のある座標は、矩形の集合で表される
- 全部回転方向が不明でも、最大で 2^50 通り
- 半分全列挙したら 2^25 通りだからもしかして通る?
  - 始点と終点から半分ずつシミュレーション
  - 矩形どうしの集合が重なっているか、Fenwick tree等を使って O(2^(N/2) log 2^(N/2)) = O(N 2^(N/2)) で判定
  - N が大きいので、そのままでは通らない

解法(2)

- 半分全列挙で列挙する矩形の個数を減らしたい
- よく考えると、奇数行目は必ず Y 方向、偶数行目は必ず X 方向の移動距離になる
- これを利用して、回転方向が ? じゃないものが連続している箇所は圧縮できる
  - 直前の ? でどっちに曲がるかによって、その後の符号が逆転するだけ

LR	?RLR	?	?LLR

解法(2)

- 圧縮した区間について、先に符号を決めてしまう
- 残りの ? を決める
  - 符号を決める問題だと思うと、X と Y は独立な問題として扱うことができる
  - それぞれ別々に半分全列挙する
- X と Y が独立になるから、高々 N/2 - a 個の符号を半分全列挙で決める問題になる
  - O((N-a) 2^((N/2-a) / 2) 2^a)) = O((N-a) 2^(N/4 + a/2)
- 最悪ケースだと a = N/2 で、やっぱり O(N 2^(N/2))

解法(AC)

- 解法(1)を使うと O(N 2^((N-a)/2)))
- 解法(2)を使うと O(N 2^(N/4 + a/2))
- a ＝ N/4 で方針を切り替えると
  - 解法(1): O(N 2^((N-N/4)/2)) ＝ O(N 2^(3N/8))
  - 解法(2): O(N 2^(N/4 + a/2)) ＝ O(N 2^(3N/8))
- つまり O(N 2^(3N/8)) で解ける!!

解法(AC)

- ログを圧縮する
- 圧縮された箇所の個数をaとして
  - a ≧ N/4 のときは解法(1)
  - a ≦ N/4 のときは解法(2)
- で符号を決める
- 最後に圧縮された箇所に符号を書き戻しつつ復元
  - 回転方向は符号の変化からわかる
  - 距離は LL_i だけ進めておいてから、ゴールに近づくように埋めていく

ジャッジ解

- 保坂 (C++) 415行
- 保坂 (Java) 490行
- 森 (C++) 641行
- 矢藤 (Java, TLE) 471行
- 古賀 (C++, 解法(1)) 369行

提出

- なし
- 実装が重いので、よっぽどの事がなければ手を出せないと思います