# Chorus 解説

川崎　理玖

# 問題概要

- A,B を N 個ずつ含む文字列がある
- 隣接文字のswapを繰り返し、以下の条件を満たしたい
- 条件：AAABBB みたいな形の部分列 K 個に分解できる
- 最小のswap回数は？

# 小課題1

- とりあえずswap 0回でOKか判定する
- ちょうど K 個ではなく K 個以下でできるか判定しても同じ
- 条件を満たす部分列に分解するとき場合の最小の個数が分かればよい

# 小課題1

- 最初に連続する A はまとめて使ってよさそう
  AAABBABABB
- バラバラに使う解を適切に変形するとまとめて使う解にできるので OK

# 小課題1

- 対応する B は先頭から選べばよさそう
  AAABBABABB
- 後ろのBを使って得することがないので
  OK

# 小課題1

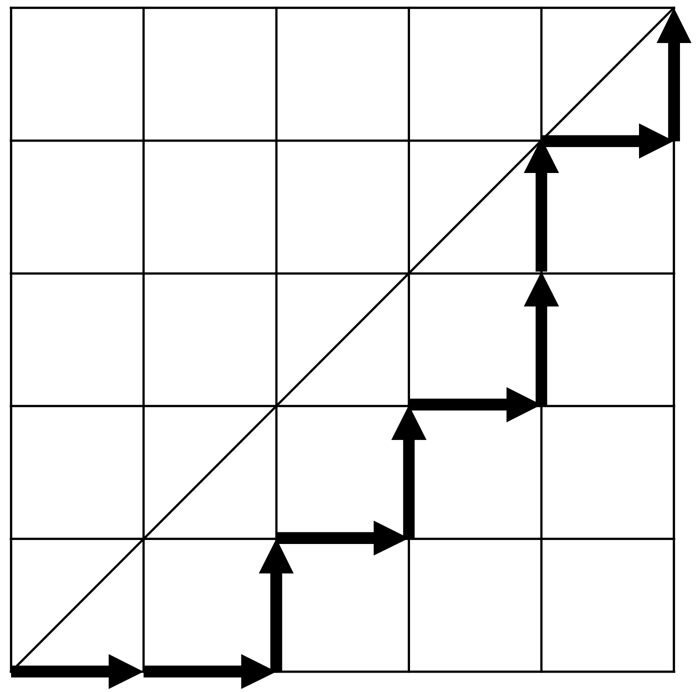
- 結局、前から貪欲に取る操作が最適
- これで分解の最小の個数が求まった

# 小課題1

- 始状態からswapを繰り返して到達できる文字列をすべて列挙し、OK なものへ到達する最小ステップ数をBFSで求める
- O(N*2^2N)で計算できる
- ちなみに判定をbitDPでひたすら頑張るとかでも一応解ける

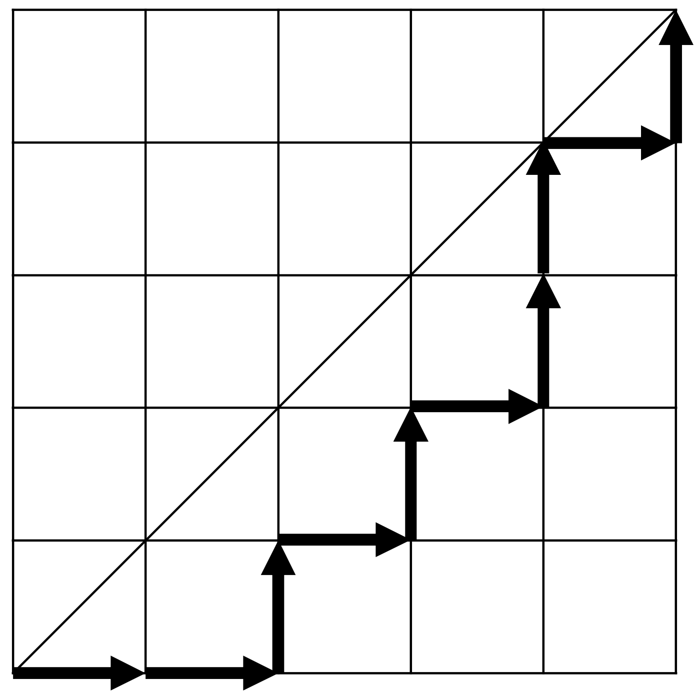
# 多項式時間解法

- 分割の個数が K 以下である、を巧みに言い換えたい
- グリッドを用意し、Aを→、Bを↑に対応させる



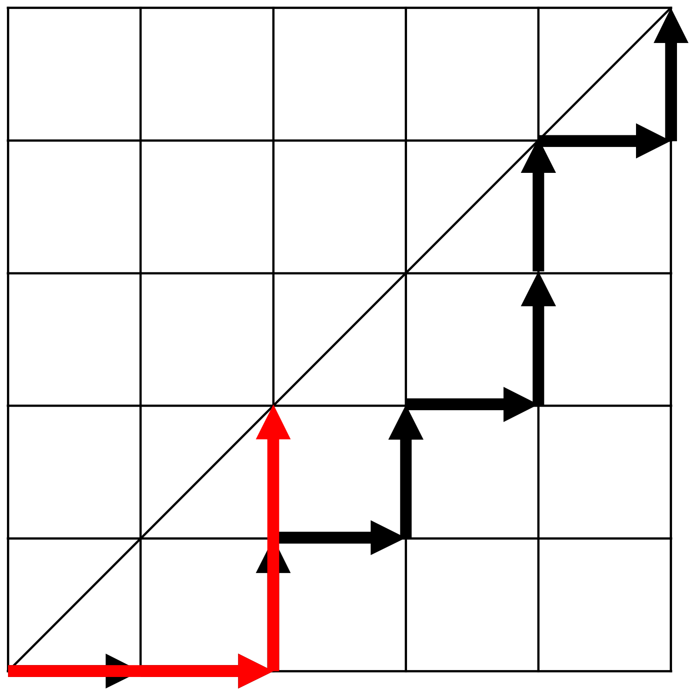
# 多項式時間解法

- さっきの貪欲を走らせてみる



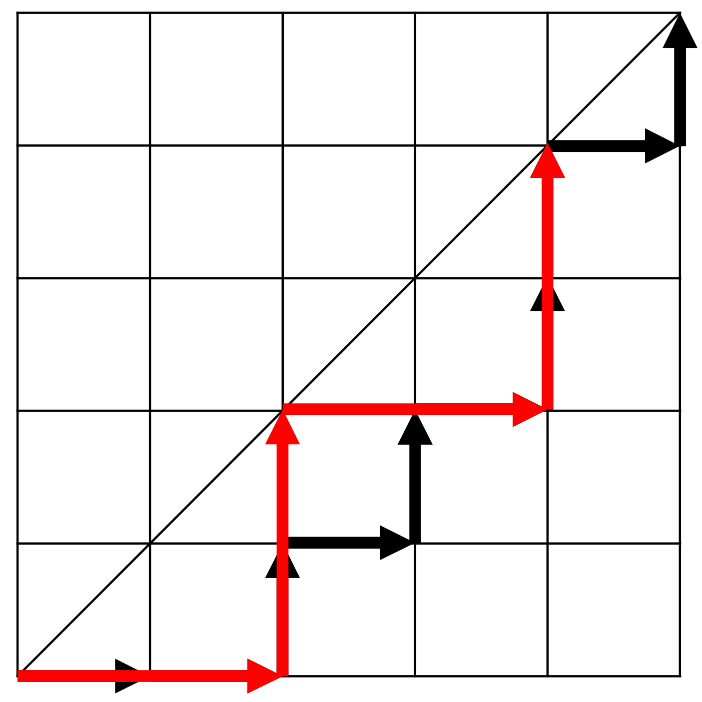
# 多項式時間解法

- さっきの貪欲を走らせてみる
- AABBをとる



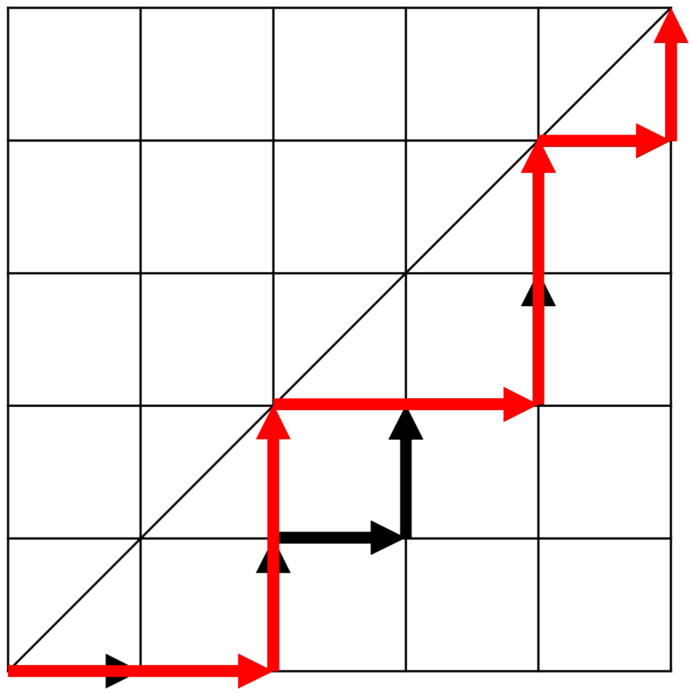
# 多項式時間解法

- さっきの貪欲を走らせてみる
- AABBをとる
- AABBをとる



# 多項式時間解法

- さっきの貪欲を走らせてみる
- AABBをとる
- AABBをとる
- ABをとる

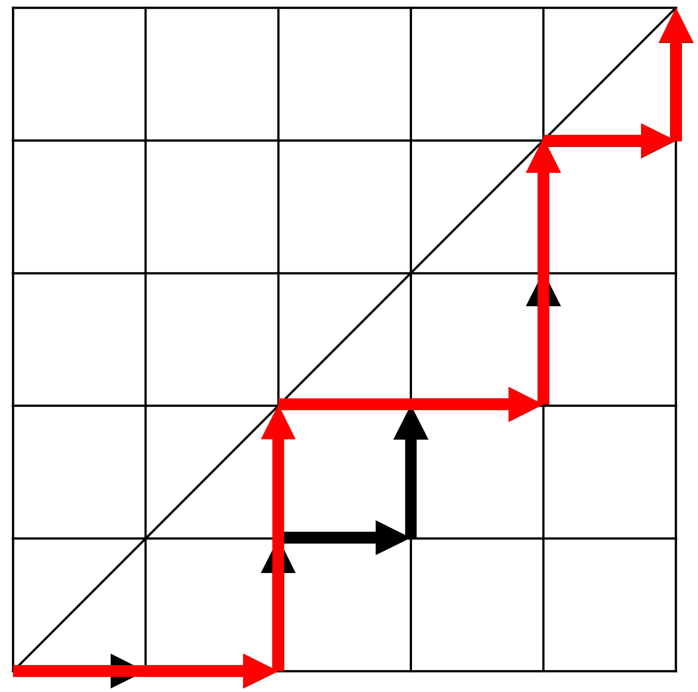


# 多項式時間解法

- 部分列 K 個に分解できる

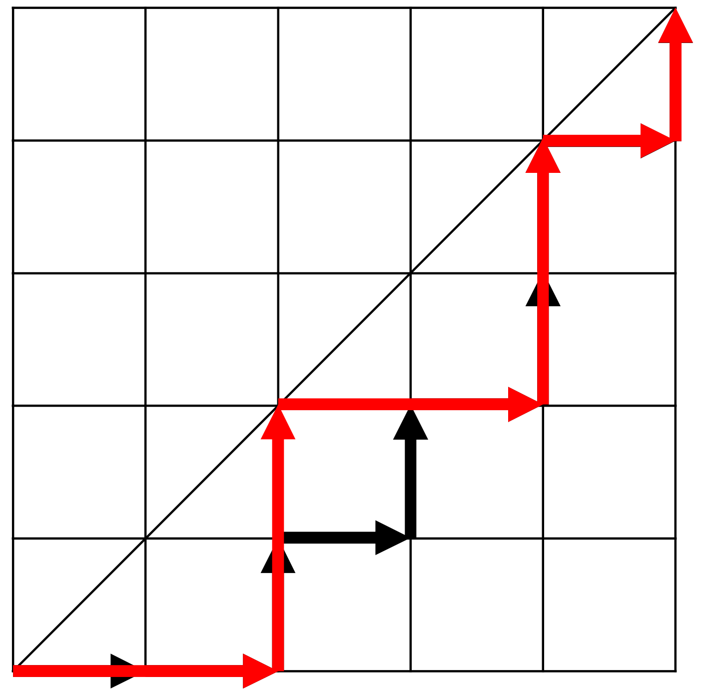
⇓

右、上、右、上、と K 回進む折れ線が右下領域と重ならない



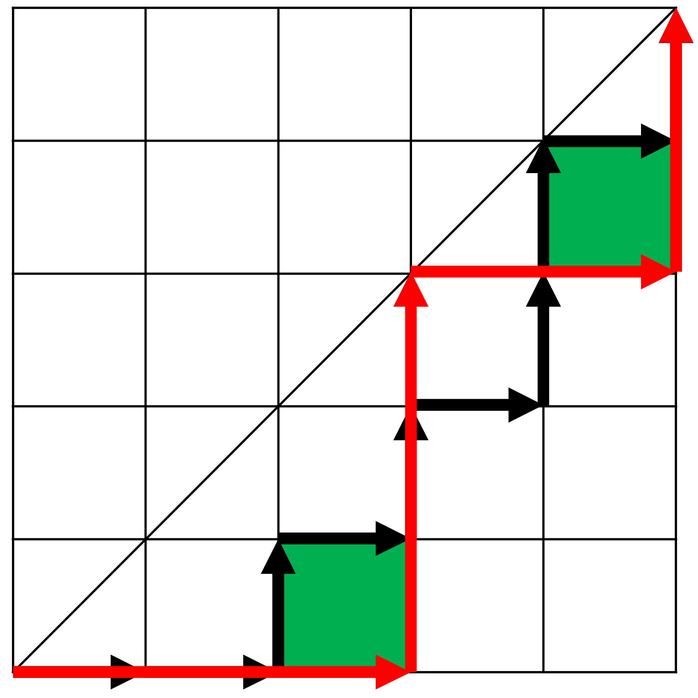
# 多項式時間解法

- 逆も成立する
- つまり、適切な折れ線があれば、それに対応する分解が作れる



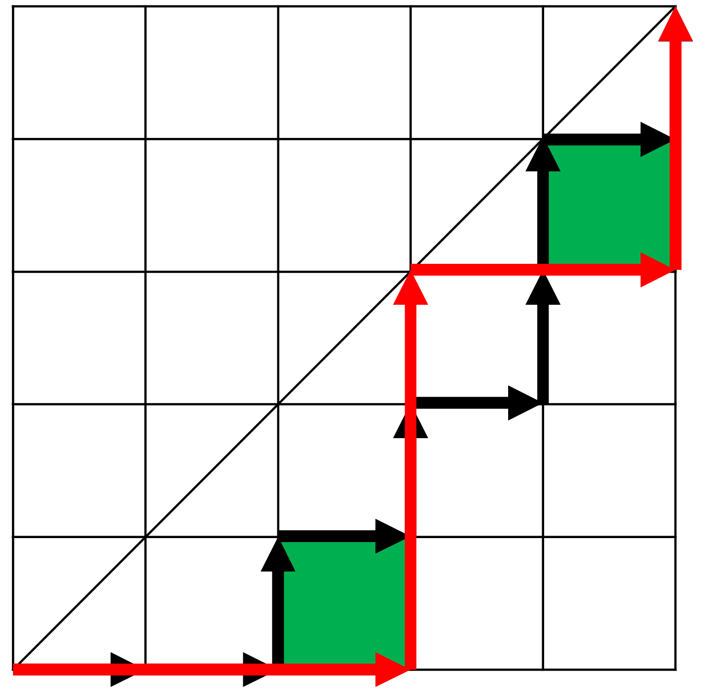
# 多項式時間解法

- ある折れ線を固定したとき、それを達成するために必要なswap回数はいくらか？
- 右下領域で折れ線の上にはみ出している面積



# 多項式時間解法

- F(L,R)を、座標 (L,L) から (R,R) まで 1 回で進む折れ線の上にはみ出す面積とする
- 例：F(0,3)=1,F(3,5)=1



# 多項式時間解法

- 問題が以下のように言い換えられる
- 0 から N までジャンプしていく
  LからRにジャンプするとコストF(L,R)
- K 回以下のジャンプの最小コストは？

# 小課題2

- F(L,R) の値は適切な前計算でO(1)
- 普通にDPすればO(N^2K)で間に合う

# 小課題3

- F(L,R)にいい性質がないと難しそう
- Monge 性がある

# 小課題3

- F(L,R+1)-F(L,R) <= F(L-1,R+1)-F(L-1,R)
- （最小化か最大化かどうかで不等号がコロコロ変わることがあるのだが、均等に刻んだ方が偏った刻み方よりもうれしい、という気持ち）
- 図示すると証明は簡単

# 小課題3

- ちょうど s 回のジャンプで場所 x にたどり着くコストを dp[s][x] として、dp[s][*] から dp[s+1][*] を求めたい
- これをある程度高速にできればOK

# 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

# 小課題3

- ちなみに SMAWK で O(N) にもなる
- 実装が面倒かつ頑張らないと定数倍が重くて分割統治に負ける
- 覚えなくてよいです

# 小課題4

- K ステップ全部求めるわけにはいかない
- コストの性質上、K ステップ以下ではなくちょうど K ステップだと思ってよい
- どうするか？

# 小課題4

- Alien DP と呼ばれるテクニックで解ける
- IOI 2016 Alien で有名になったことが名前の由来
- これも各自で調べてください

# 小課題4

- 適当な定数 C に対し、G(L,R)＝F(L,R)+C とおき、ジャンプのコスト G の問題を解きたい
- 分割統治（オンラインーオフライン変換）+Monotone Minima で O(N log^2 N)

# 小課題4

- 答えが O(N^2) なので、ラグランジュ緩和の二分探索の分で O(log N) 回走る
- 全体で O(N log^3 N)
- Monotone Minima の log が軽いので通る

# 小課題5

- 内側のDPパートをさらに高速化する
- LARSCHを使うと O(N) （全体 O(N log N)）で AC!
- 実装が面倒だしそこまで速くない
- これも覚えなくてよいです

# 小課題5

- 簡易LARSCHなるものがある（https://noshi91.hatenablog.com/entry/2023/02/18/005856）
- 実装が簡単
- O(N log N) だけど定数倍がよい
- 全体 O(N log^2 N) で AC!

# 小課題5

- 何も知らないとどうなるのか？
- CHT とスライド最小値で DP を高速化すれば O(N)
- 全体 O(N log N) で AC!

# 得点分布

- 0点　：4 人
- 16点　：6 人
- 40点　：12 人
- 61点　：3 人
- 87点　：1 人
- 100点　：3 人