

立命合宿2017 Day3

G: 恋のジュンレッツRun run run!

原案: 井上

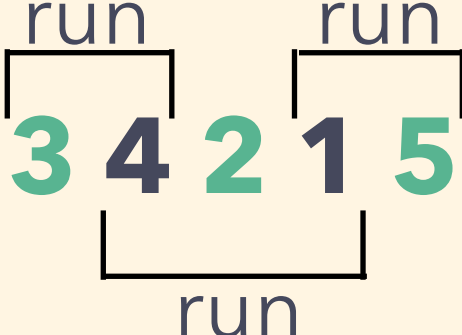
問題文: 井上

テスター: 井上・栗田・杉江

解説: 井上

問題概要

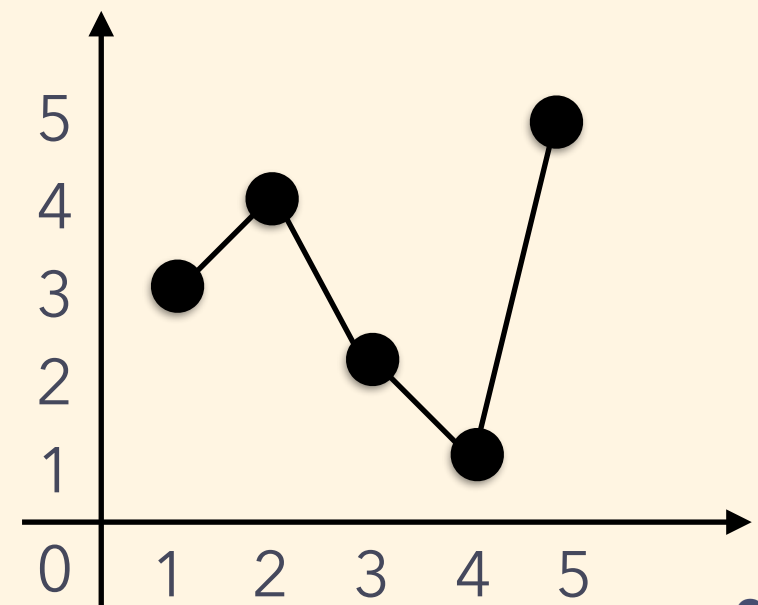
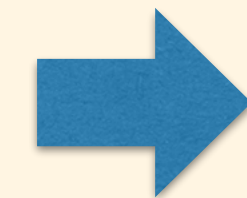
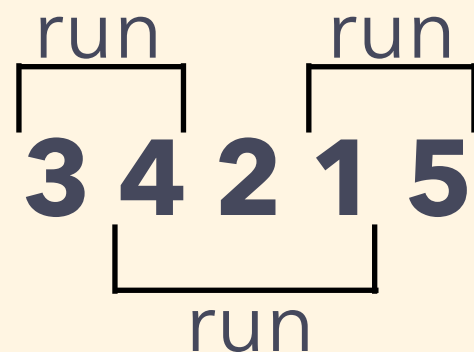
- 入力: テキスト順列 q , パターン順列 p
- 出力: p が q にマッチするか否か?
- マッチの定義: q の部分列が p と同じ相対順序
- 制約: $1 \leq |p|, |q| \leq 10^5$, q の run はちょうど 3 つ
- 順列の run: 単調増加 or 減少な極大連続部分列

テキスト:  3 4 2 1 5

パターン: 2 1 3

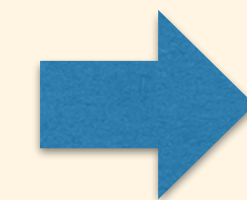
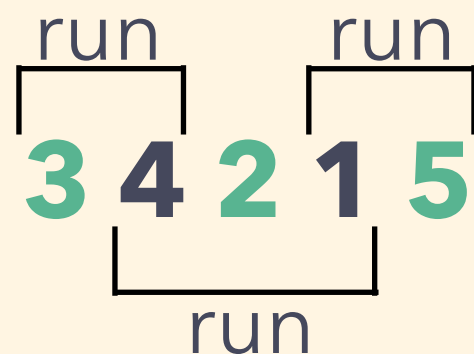
順列行列

- パターンを考えるとき、 permutation matrix 的なもの考えるとイメージがわかりやすい
- 2次元座標に順列をマッピング
- i 番目の要素が p_i のとき、 (i, p_i) に点がある
- run の端点: 山や谷に対応

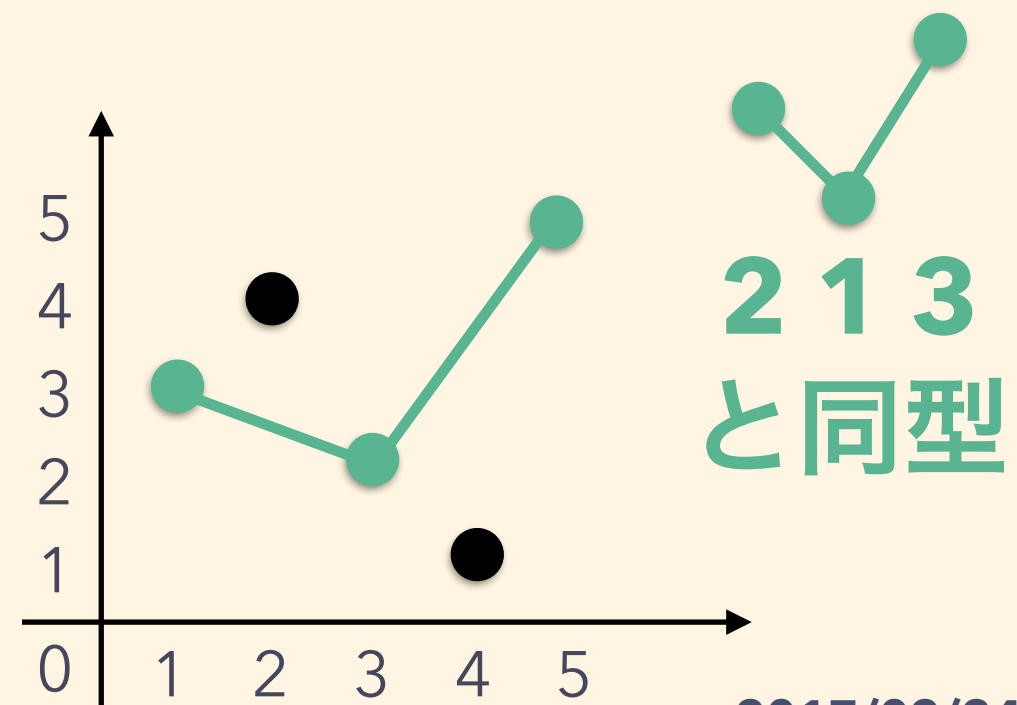


順列行列

- パターンを考えるとき、 permutation matrix 的なもの考えるとイメージがわかりやすい
- 2次元座標に順列をマッピング
- i 番目の要素が p_i のとき、 (i, p_i) に点がある
- run の端点: 山や谷に対応

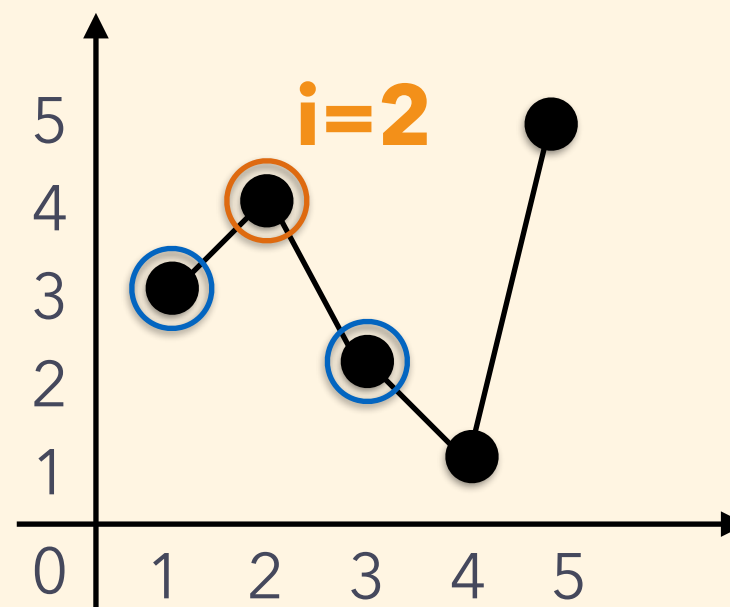


4



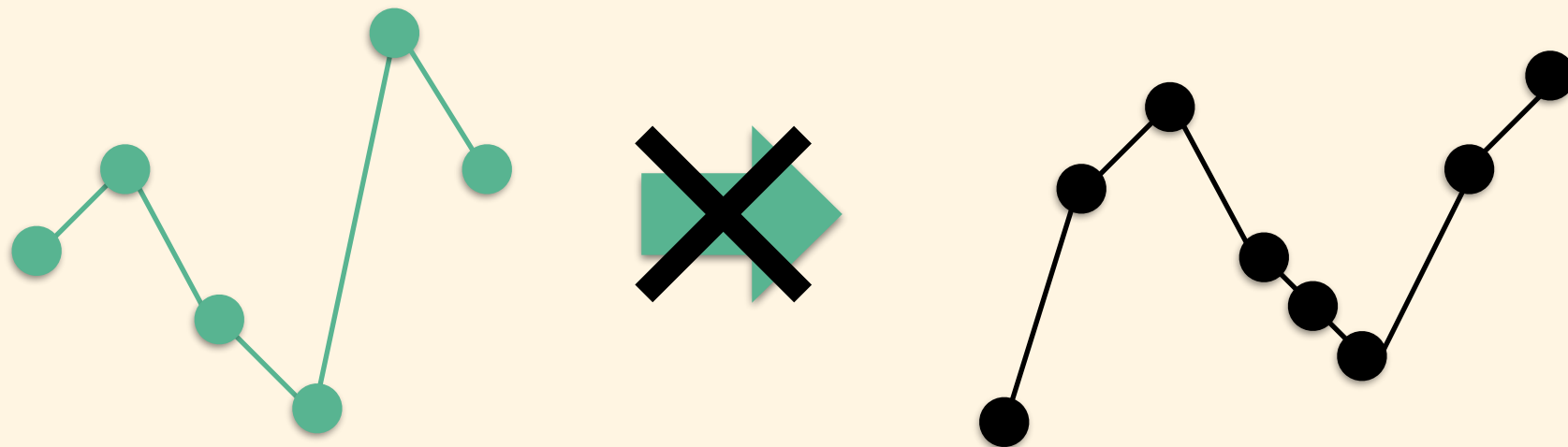
run の計算

- run (の端点) をあらかじめ計算しておく と 楽
- 線形時間で求められる
- $p_{i-1} < p_i > p_{i+1}$ (山) か $p_{i-1} > p_i < p_{i+1}$ (谷) か
を各 i についてチェックすればよい



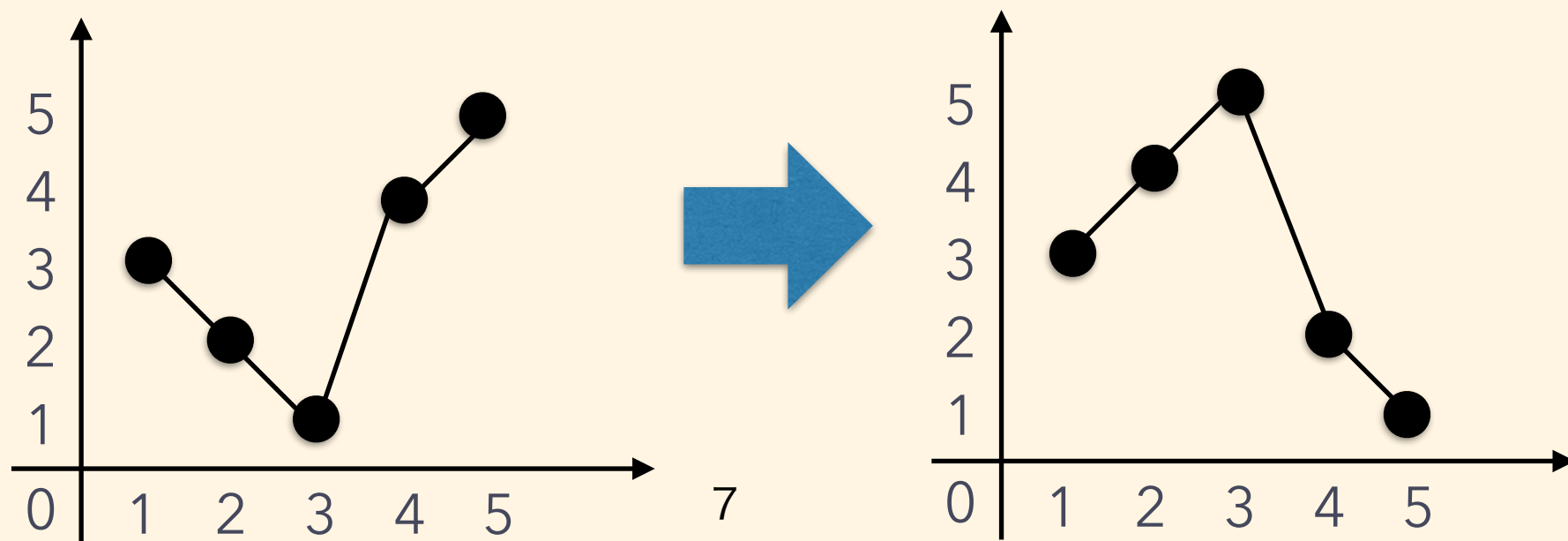
パターンの run

- 制約: テキスト q の $\text{run}(q) = 3$
- ではパターン p の run は?
- $\text{run}(p) > 3$ なら絶対に q にマッチしないことがすぐわかる $\Rightarrow \text{run}(p) \leq 3$ と仮定してよい



マッチ結果を変えない変更

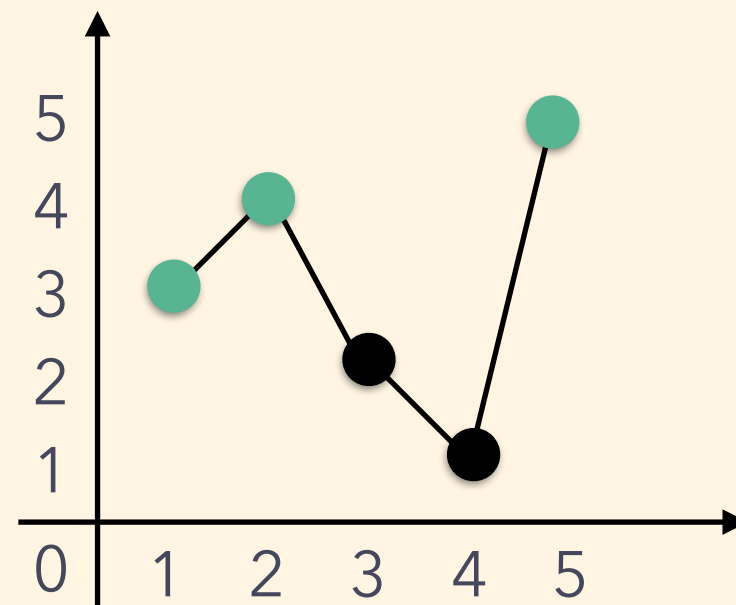
- パターン p は大まかに $\{\text{run}(p) = 1, 2, 3\} \times \{\text{最初の run が昇 or 降}\}$ の6通りに分類できる
- p が降始まりの場合、 p, q 共に全要素 a を $|p|+1-a$ ($|q|+1-a$) に変更してもマッチ結果は変わらない
 - y 軸方向に反転するイメージ
 - 以降、この前処理で p は常に昇始まりと仮定



run(p) = 1 の場合

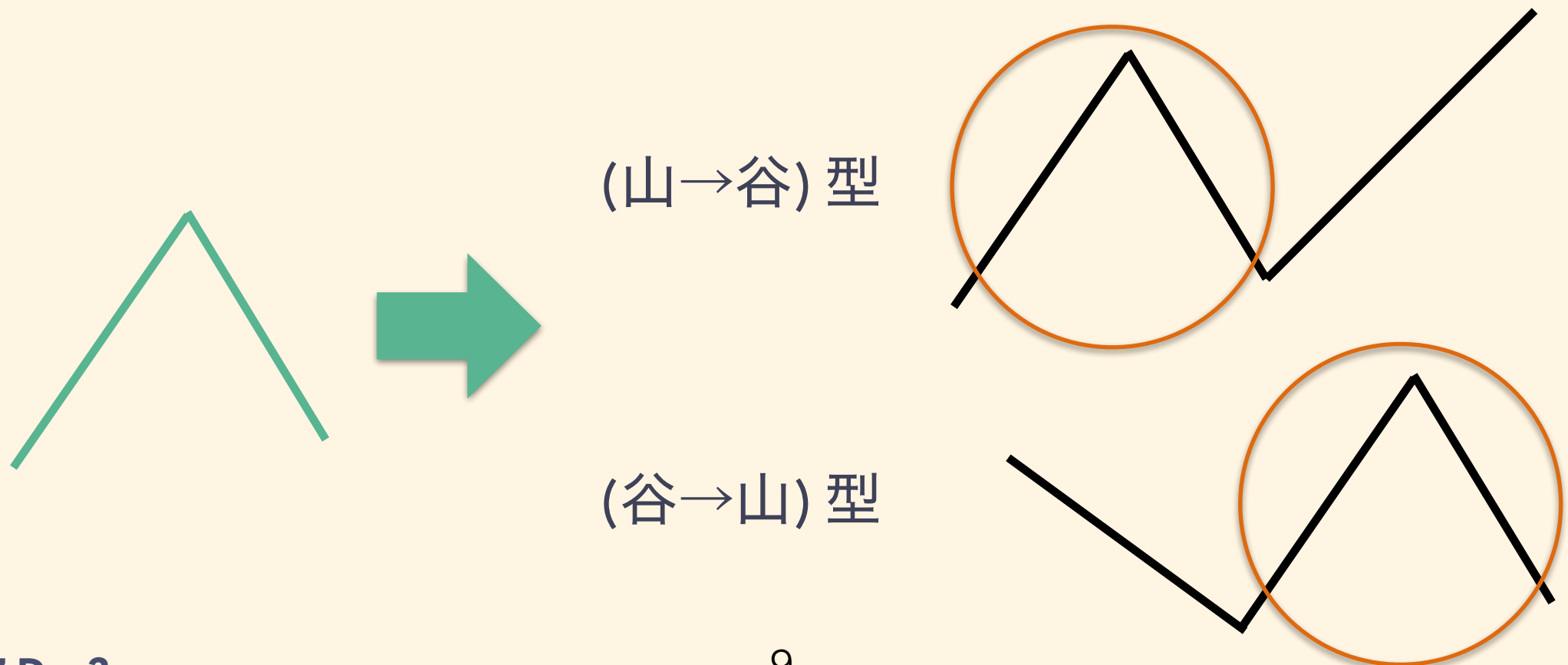
- p は増加列 \Rightarrow q の最長増加部分列 (いわゆる LIS) の長さが $|p|$ 以上なら Yes, 未満なら No
- LIS は $O(|q| \log |q|)$ で求められる (c.f. 蟻本)
- (参考) $\text{run}(q) = 3$ を利用すると線形時間でも計算可能

パターン: 1 2 3



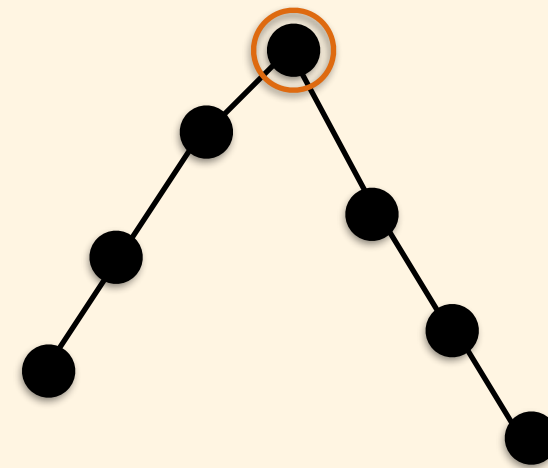
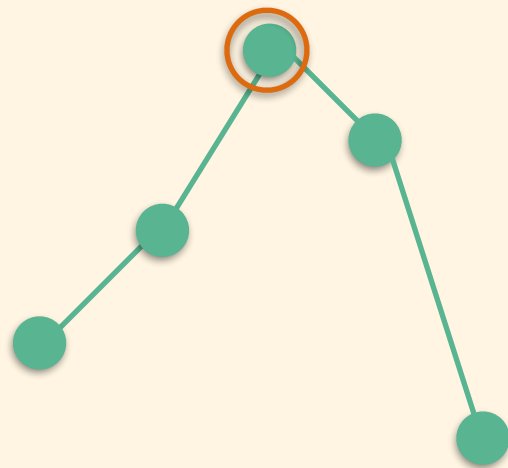
run(p) = 2 の場合

- p は山型、q は (山→谷) 型か (谷→山) 型かの形
- p が q にマッチ \Leftrightarrow q の山型部分だけとマッチ
- $\text{run}(p) = \text{run}(q) = 2$ と思ってよい



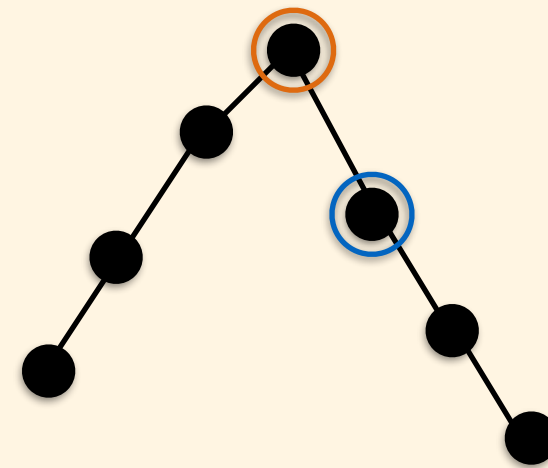
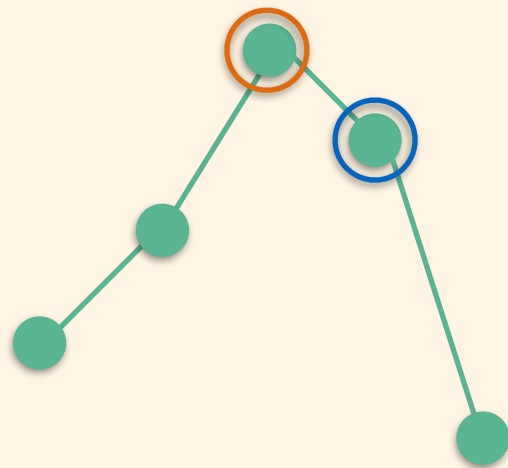
$\text{run}(p) = \text{run}(q) = 2$ の場合

- ピークは揃えてよい
- 残りは左右に降っていく、 p でピークに近い方から合わせるように q も同時に降る
- 1 個前にとった数より小さければよく、できれば大きい方が嬉しいので、貪欲にとればOK



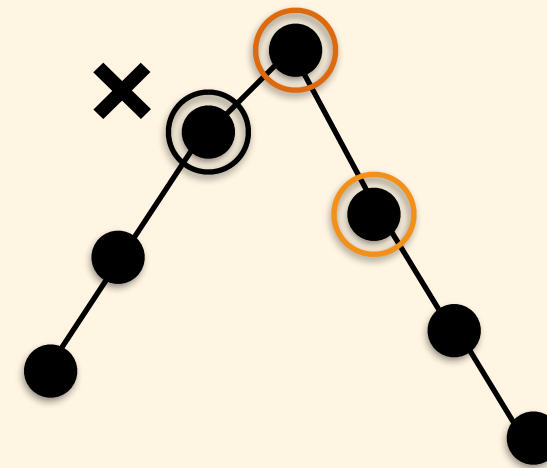
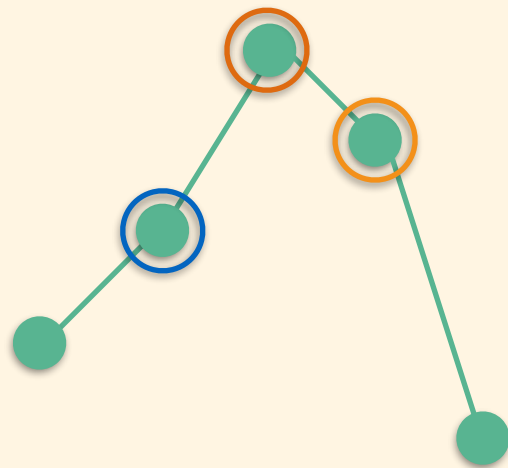
$\text{run}(p) = \text{run}(q) = 2$ の場合

- ピークは揃えてよい
- 残りは左右に降っていく、 p でピークに近い方から合わせるように q も同時に降る
- 1 個前にとった数より小さければよく、できれば大きい方が嬉しいので、貪欲にとればOK



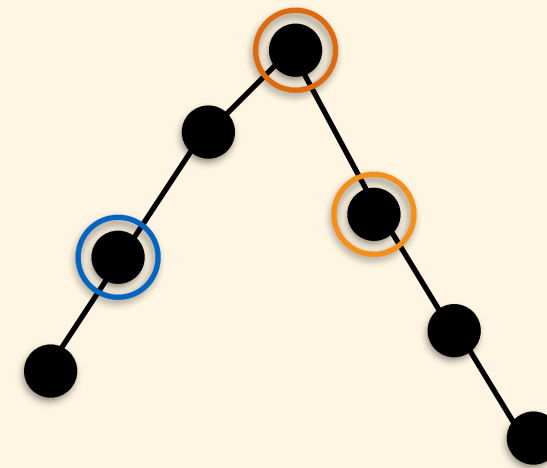
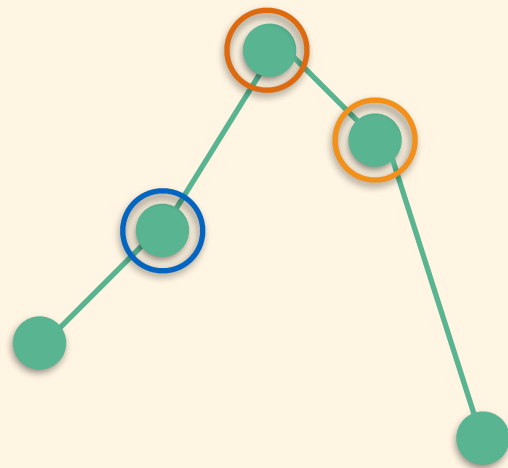
$\text{run}(p) = \text{run}(q) = 2$ の場合

- ピークは揃えてよい
- 残りは左右に降っていく、 p でピークに近い方から合わせるように q も同時に降る
- 1 個前にとった数より小さければよく、できれば大きい方が嬉しいので、貪欲にとればOK



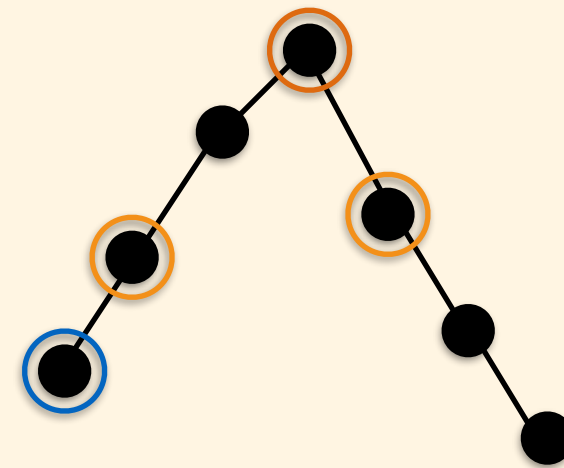
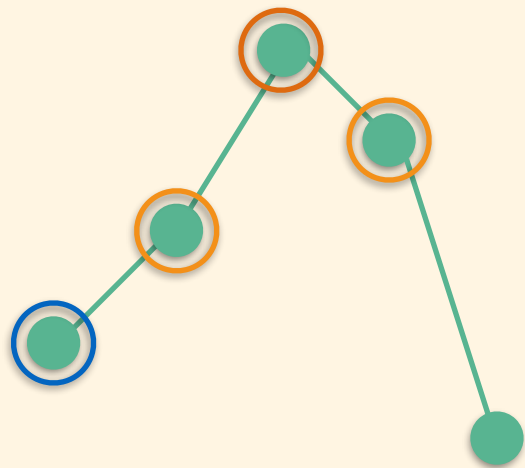
$\text{run}(p) = \text{run}(q) = 2$ の場合

- ピークは揃えてよい
- 残りは左右に降っていく、 p でピークに近い方から合わせるように q も同時に降る
- 1 個前にとった数より小さければよく、できれば大きい方が嬉しいので、貪欲にとればOK



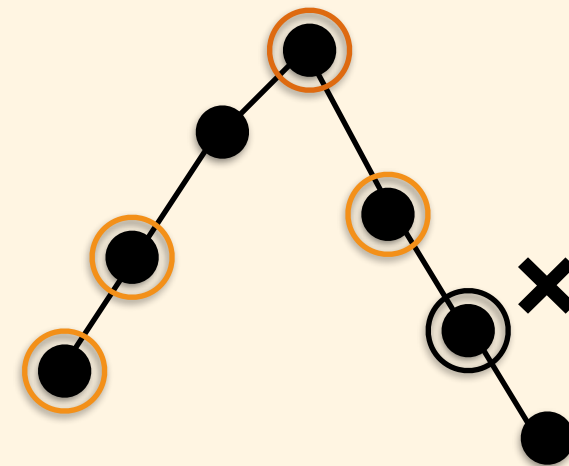
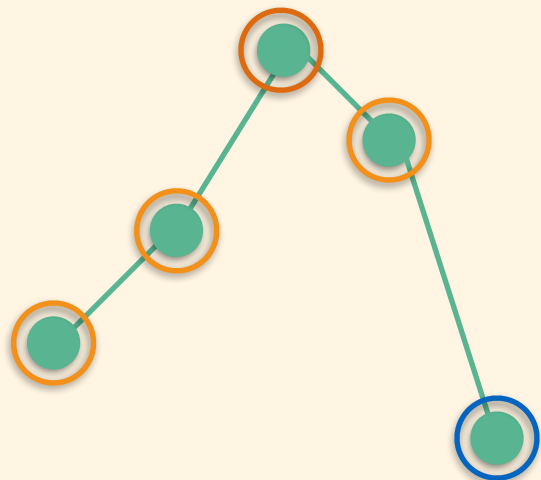
$\text{run}(p) = \text{run}(q) = 2$ の場合

- ピークは揃えてよい
- 残りは左右に降っていく、 p でピークに近い方から合わせるように q も同時に降る
- 1 個前にとった数より小さければよく、できれば大きい方が嬉しいので、貪欲にとればOK



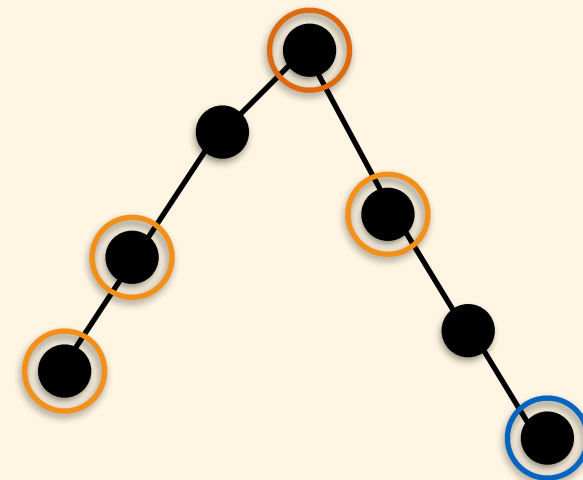
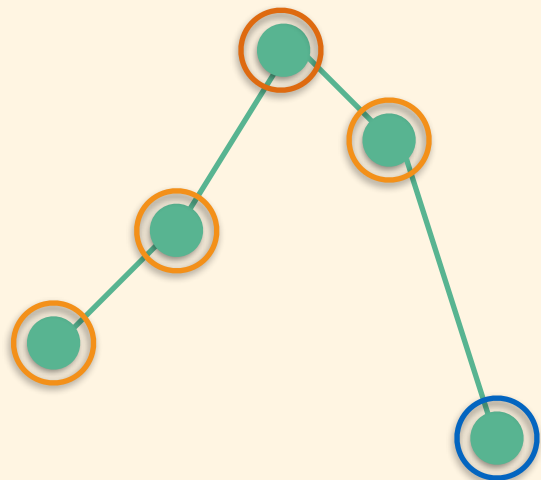
$\text{run}(p) = \text{run}(q) = 2$ の場合

- ピークは揃えてよい
- 残りは左右に降っていく、 p でピークに近い方から合わせるように q も同時に降る
- 1 個前にとった数より小さければよく、できれば大きい方が嬉しいので、貪欲にとればOK



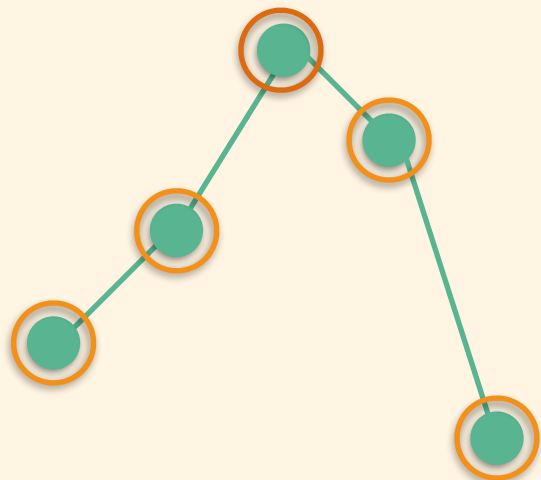
$\text{run}(p) = \text{run}(q) = 2$ の場合

- ピークは揃えてよい
- 残りは左右に降っていく、 p でピークに近い方から合わせるように q も同時に降る
- 1 個前にとった数より小さければよく、できれば大きい方が嬉しいので、貪欲にとればOK

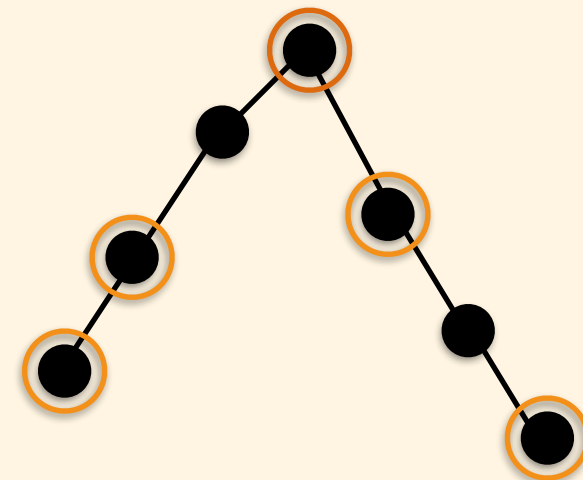


$\text{run}(p) = \text{run}(q) = 2$ の場合

- ピークは揃えてよい
- 残りは左右に降っていく、 p でピークに近い方から合わせるように q も同時に降る
- 1 個前にとった数より小さければよく、できれば大きい方が嬉しいので、貪欲にとればOK

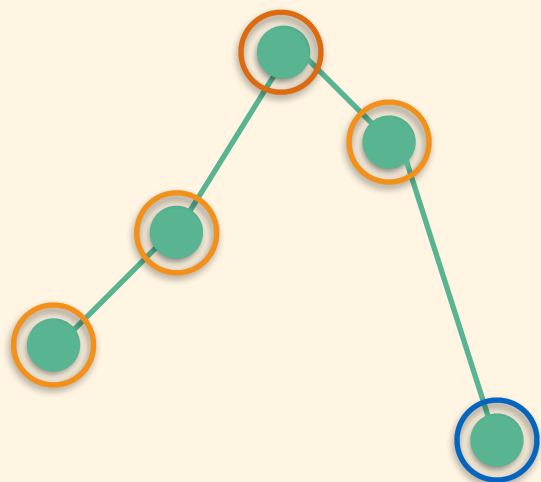


Yes!

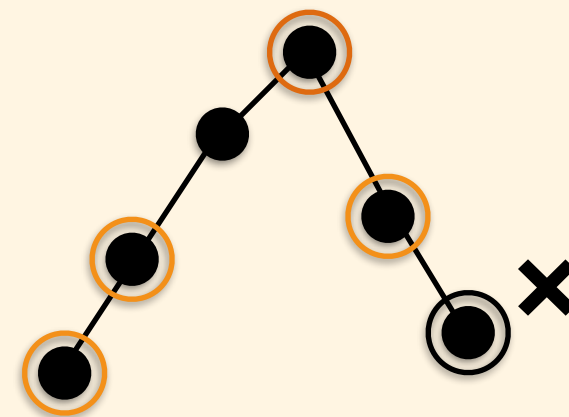


$\text{run}(p) = \text{run}(q) = 2$ の場合

- ピークは揃えてよい
- 残りは左右に降っていく、 p でピークに近い方から合わせるように q も同時に降る
- 貪欲な取り方で失敗したら No、最後までいけたら Yes

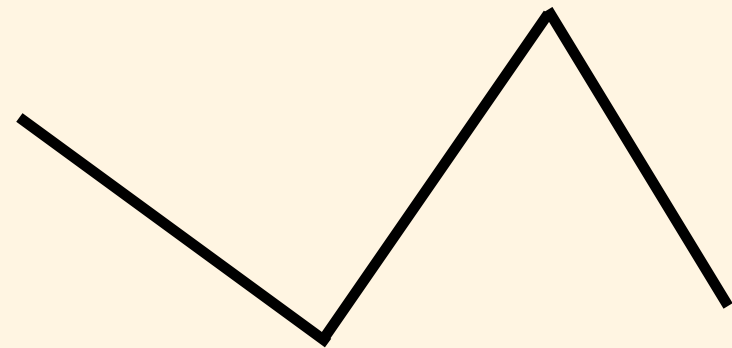
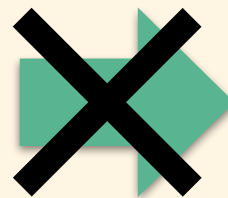


No...



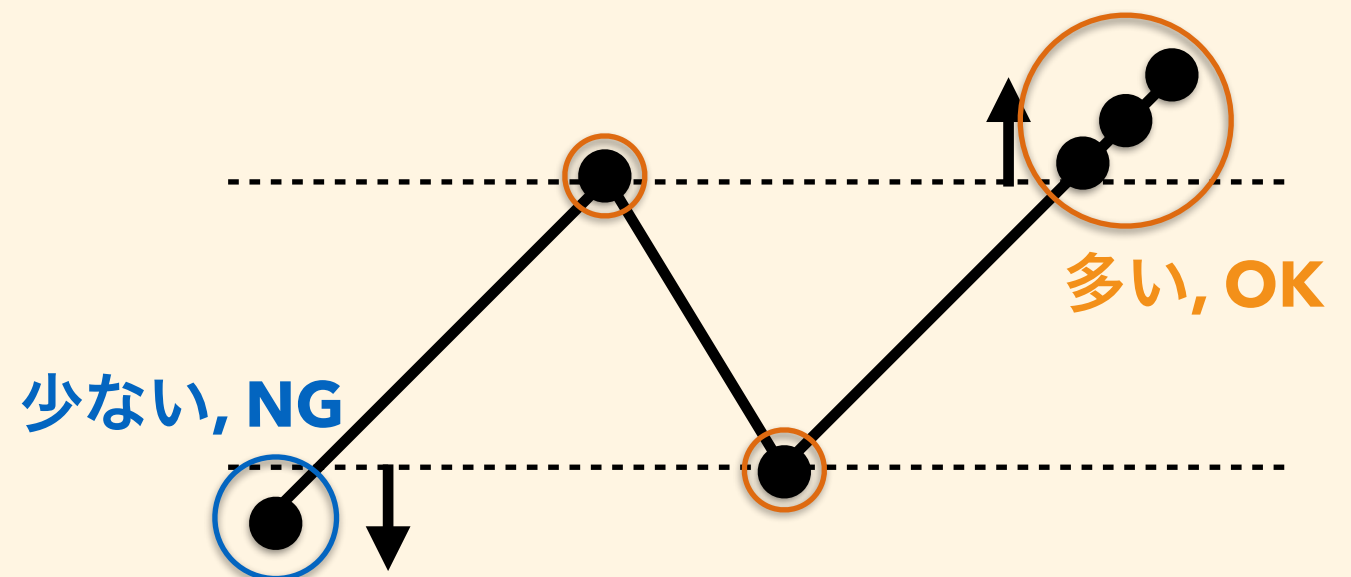
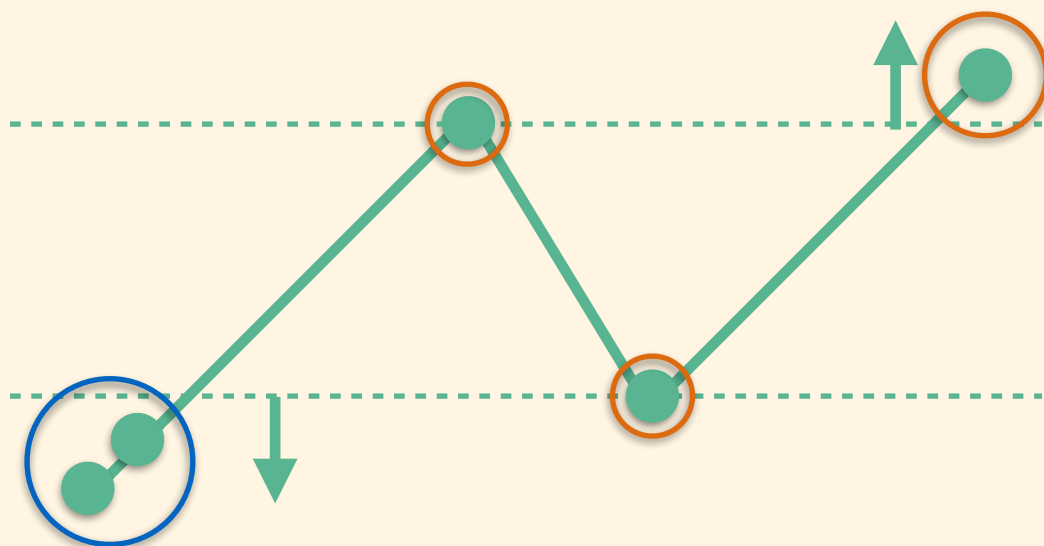
run(p) = 3 の場合

- p は (山→谷) 型なので、q も (山→谷) 型でなければ絶対に No
- 一致していれば、run(p) = 2 の場合と似た貪欲ができる



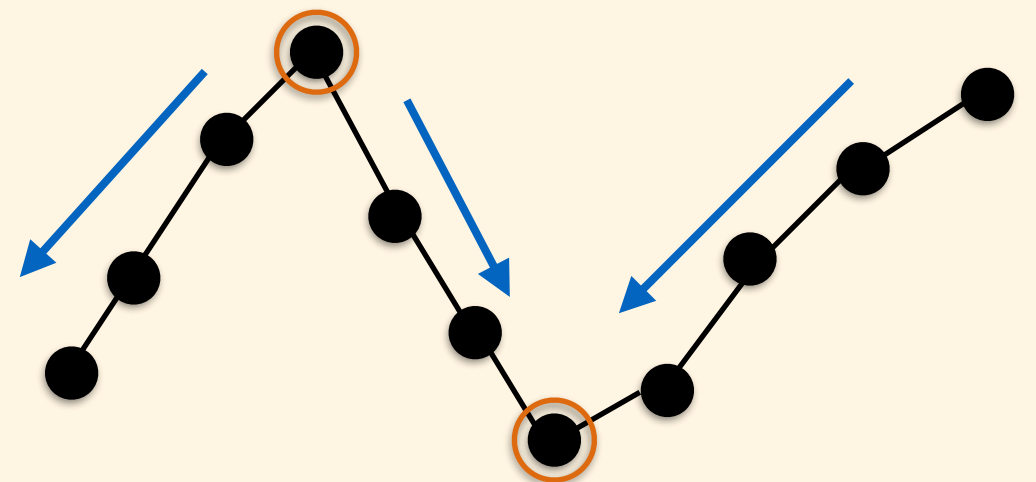
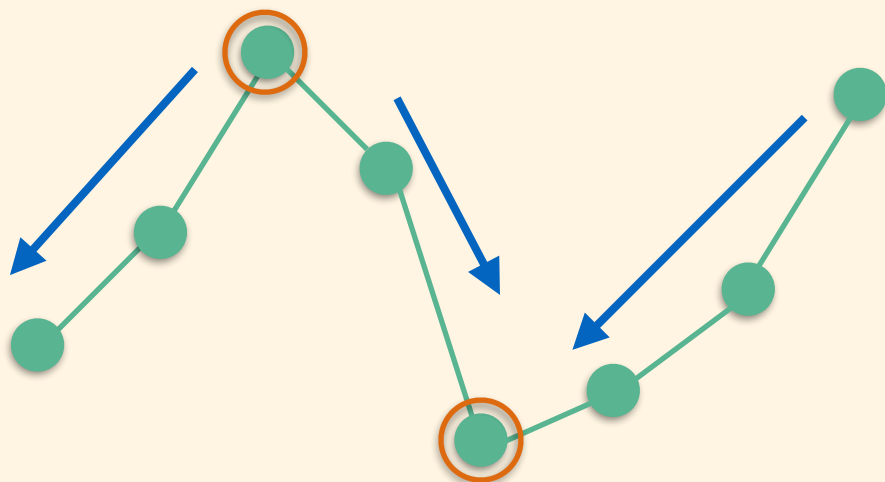
$\text{run}(p) = \text{run}(q) = 3$ の場合

- 同じくピークは揃える
- 左右の端にピークより小さい/大きい要素がありうるので、あらかじめ除いておくとうわかりやすい
- p, q それぞれの上記を満たす要素数を左右ごとに比べ、 q の方が少なかったら No とするだけ



$\text{run}(p) = \text{run}(q) = 3$ の場合

- 同じくピークは揃える
- 山の左右と右端の 3 箇所を見ながら、 p で最大の要素のポインタを q の方でも動かす
- $\text{run} = 2$ と同じように、 p に合わせながら条件を満たす最大のものを取り続ける貪欲で OK



余談

- Run が短いテキスト順列に対する FPT アルゴリズムがある
 - Marie-Louise Bruner and Martin Lackner,
"A Fast Algorithm for Permutation Pattern Matching
Based on Alternating Runs," Algorithmica, 2016
 - p の各 run と q の連続した runs の対応として、ありうるものを全部試す
 - DPでその対応を満たすマッチがあるか判定
- 今回は run がさらに短く制限されているため、run の対応が1通りしかない & DP せずとも貪欲で答えがわかる

Writer 解

- 井上: 134 lines, 3171 bytes (C++)
- 栗田: 134 lines, 3268 bytes (C++)
- 杉江: 171 lines, 5120 bytes (C++)

統計

- Accept / Submission
 - 16.67% (2/12)
- First Acceptance
 - On-site: public_latte_mcdwl (02:47)
 - On-line: satanic0258 (01:45)