

## Problem

Given a permutation, reorder as few elements as possible to make it mountain-shaped (first increasing then decreasing) in  $\mathcal{O}(n \log n)$ .

## Problem

Given a permutation, reorder as few elements as possible to make it mountain-shaped (first increasing then decreasing) in  $\mathcal{O}(n \log n)$ .

## First insight

If we place elements from smallest to largest, each element will go to one of the borders. This leads to a quadratic DP with state space

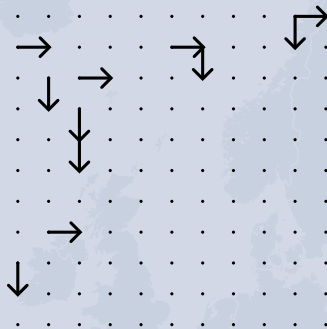
(Num elements placed on the left so far, Num elements placed on the right so far).

Transition value is 0 if element is misplaced, and 1 if it stays where it is.

### Visualizing the DP state space

Need to find path from top left to bottom right that

- only moves down and right, and
- hits as many arrows as possible.

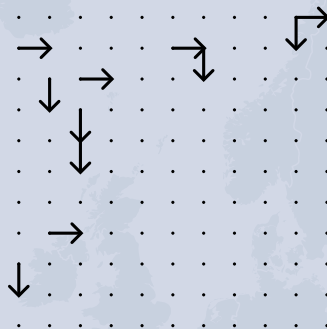


## Visualizing the DP state space

Need to find path from top left to bottom right that

- only moves down and right, and
- hits as many arrows as possible.

Key insight: there are at most  $2n$  arrows!



## Visualizing the DP state space

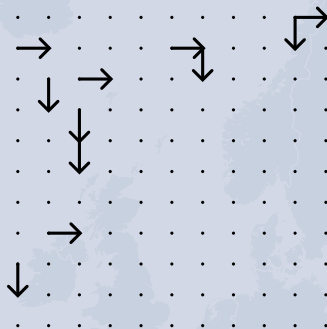
Need to find path from top left to bottom right that

- only moves down and right, and
- hits as many arrows as possible.

Key insight: there are at most  $2n$  arrows!

Possible approaches:

- Rephrase DP as Longest Increasing Subsequence.
- Quickly simulate DP using segment tree or ordered set.



Need to find path from top left to bottom right that

- only moves down and right, and
- hits as many arrows as possible.

Key insight: there are at most  $2n$  arrows!

### Possible approaches:

- Rephrase DP as Longest Increasing Subsequence.
- Quickly simulate DP using segment tree or ordered set.

