

# Paint

**Giri Narasimhan**

Programming Team

Fall 2018

# A. Product Sum

- Given n numbers  $A[1..n]$
- Characteristic of array  $c = \sum_{i=1}^n a_i \cdot i$
- Allowed one change
  - Move one item from its current location to a different location
- Find the move that maximizes resulting c

# Properties

- If an item  $x$  is moved from location to its right by  $k$  positions and if there are no items larger than it on the way, then the increase is proportional to the **difference** between  $kx$  and the **sum of the items it passes**.

# B. Iorha Loves Strings

# C. ACM Rank

- Data contains a stream of requests
  - **S** minute teamID problemID result
  - **R** teamID
  - **T** rank
- Goal is to answer each query as efficiently as possible

# Augmented RB Tree

- As with Rank and Select,
  - Augment RB tree with size (of subtree) info
- **S** translates to
  - insert or update operation
- **R** translates to
  - inorder tree traversal
- **T** translates to
  - Doing select operation on augmented tree

# E. Tree Augmentation

- Examples
  - Tree -> AugTree
- Examples
  - AugTree -> Tree
  - Challenges

# Observation

- Every vertex forms a **star** with its neighbors
- Every **star** in Tree becomes a **clique** in the AugTree
- Thus identify the cliques
- Is that enough
- What about adjacent neighbors in Tree?
  - Cliques with common vertices



# Algorithm for Augmented Tree?

- Use observations above to design an algorithm