

## IOI'12: Jousting Tournament Solution

Solutions of various ranges of complexity are possible: all of them are essentially based on the trivial idea of trying all possible positions and simulating the tournament.

A trivial way to do that takes time  $O(N^3)$ . We hereby describe a  $O(N^2)$ -time solution.

- The whole tournament can be thought of as a tree whose leaves represent the knights and where all other nodes represent the winner of a round. The structure of the tree is the same regardless of where we put ourselves in the initial rank, only labels of the nodes (i.e., round winners) change.
- The latter tree leads to an  $O(N^2)$  solution: the tree construction takes time  $O(N^2)$ ; then for each of the possible  $N$  positions, we have to determine how far up our knight can go, so we are left with another  $O(N^2)$  checks.

To go down to  $O(N \log N)$  time we need to optimize the tree construction and the tournament's simulations.

- To speed up the tree construction to  $O(N \log N)$ , we can use a binary range tree to get quickly the knight that is currently in any given challenged position.
- To speed up the second phase, we make two observations.
  1. Let us call "white"/"black" the knights that are weaker/stronger than the late knight. Then, when we place the late knight in a certain position, we have to determine how far up we can go without finding a node that has some black leaf below it. For example, if we decide to place the late knight in the leftmost position, we want to find the leftmost node that has the longest path to a leaf and contains only white leaves under it.
  2. Every time we place the knight in a given position, we are simply shifting (to the left) every knight to his left.
- Combining these two observations, we don't need to actually try a position to see how far up we can go: it is sufficient to proceed as described in the first observation but allowing the leftmost descendant to be black, and requiring only the remaining ones to be white.
- Doing it in this way, the second phase is  $O(N)$  because of the second observation.