

IZBORI

For each pair of candidates (A, B) we examine all the voters' lists. The candidate preferred by more voters receives a point.

TURBO

Trying to build the cycle on the fly or keeping track of it implicitly can be painful. A simpler solution builds the cycle into a auxiliary array every time the T key is pressed for the first time. On further presses of the T key we just print numbers from the constructed cycle.

The time complexity is $O(N^2)$, because constructing the cycle takes $O(N)$ time.

KULE

Let A_{ij} be the given matrix. Let $\text{row}(x)$ and $\text{col}(x)$ be the sums of the x -th row and column, respectively.

Let the positions of the rooks be $(r1, c1)$ and $(r2, c2)$. There are three distinct cases:

- 1) The rooks are in the same row ($r1 = r2$)
- 2) The rooks are in the same column ($c1 = c2$)
- 3) Neither of the above

Cases 1 and 2 can be solved trying all possibilities (there are three variables). The sum can then be calculated in $O(1)$ using the identities above.

For the third case, consider what the formula would be if we knew the final positions of both rooks:

$$\text{Sum} = \text{row}(r1) + \text{row}(r2) \quad (1)$$

$$+ \text{col}(c1) - A[r2][c1] - 2*A[r1][c1] \quad (2)$$

$$+ \text{col}(c2) - A[r1][c2] - 2*A[r2][c2] \quad (3)$$

In the above formula, we subtract numbers we'd included once or twice.

If we try all possible ways of choosing $r1$ and $r2$, we can then independently choose different columns $c1$ and $c2$ such that the parts (2) and (3) of the above formula are maximized.

The complexity for all three cases is thus $O(N^3)$.