Write a function `solution` that, given an array A of N integers, returns the largest integer K > 0 such that both values K and −K (the opposite number) exist in array A. If there is no such integer, the function should return 0.

Examples:

1. Given A = [3, 2, −2, 5, −3], the function should return 3 (both 3 and −3 exist in array A).
2. Given A = [1, 1, 2, −1, 2, −1], the function should return 1 (both 1 and −1 exist in array A).
3. Given A = [1, 2, 3, −4], the function should return 0 (there is no such K for which both values K and −K exist in array A).

Write an **efficient** algorithm for the following assumptions:

- N is an integer within the range [1..100,000];
- each element of array A is an integer within the range [−1,000,000,000..1,000,000,000].