Poisonous nim (`nim`)

You are playing a game of *Poisonous Nim* against Elia. Poisonous Nim is played by two players using N piles of coins. The piles are numbered from 0 to $N-1$ and the i-th pile initially contains A_i coins.

The players alternate in making moves: a move consists in choosing a non-empty pile and removing any number of coins between one and the current number of coins in the pile from the chosen pile.

However, whenever a player removes coins from a pile, the pile becomes poisonous, and it cannot be chosen by the opponent in his next move. Note that a pile only remains poisonous for one move.

The first player who cannot make any move loses.

You are going first, and want to find out how many different starting moves will guarantee you victory assuming both you and Elia play optimally. Two moves are considered different if they remove coins from different piles or if the number of removed coins is different.

Implementation

You will have to submit a single file, with a `.cpp` extension, with the following function.

C++	`long long play(int N, vector<int> A);`

- The function will be called once during the execution of your program.
- The integer N is the number of piles of coins.
- The array A contains the number of coins in each pile.
- The function must return the number of starting moves that guarantee you victory.

Sample Grader

There is a sample grader attached, which reads from `stdin` and writes to `stdout` according to the following format:

The input file is made up of 2 lines, containing:

- Line 1: N.
- Line 2: N integers $A_0...A_{N-1}$.

The output file is made of one line, containing the value returned by the function `play`.

Constraints

- $2 \le N \le 1\,000\,000$
- $1 \le A_i \le 1\,000\,000\,000$, for each i from 0 to $N-1$.

Scoring

- **Subtask 0 [0 points]**: Samples
- **Subtask 1 [11 points]**: $N = 2$
- **Subtask 2 [8 points]**: $N \le 4, A_i \le 50$
- **Subtask 3 [19 points]**: $N \le 100, A_i \le 100$
- **Subtask 4 [12 points]**: $N \le 100, A_i \le 10\,000$
- **Subtask 5 [17 points]**: $N \le 1000, A_i \le 10\,000$
- **Subtask 6 [11 points]**: $N \le 1000$

- **Subtask 7 [9 points]:** $N \leq 100\,000$
- **Subtask 8 [13 points]:** No additional constraints.

Examples

stdin	stdout
3 1 5 1	4
3 4 6 5	6

Explanation

In the **first sample case** there are 4 starting moves that guarantee you victory (note that the piles are indexed starting from 0):

- removing 1 coin from pile with index 1.
- removing 2 coins from pile with index 1.
- removing 3 coins from pile with index 1.
- removing 5 coins from pile with index 1.

In the **second sample case** there are 6 starting moves that guarantee you victory.