

[HOME](#) [TOP](#) [CATALOG](#) [CONTESTS](#) [GYM](#) [PROBLEMSET](#) [GROUPS](#) [RATING](#) [EDU](#) [API](#) [CALENDAR](#) [HELP](#)
[PROBLEMS](#) [SUBMIT CODE](#) [MY SUBMISSIONS](#) [STATUS](#) [HACKS](#) [ROOM](#) [STANDINGS](#) [CUSTOM INVOCATION](#)

F. Peanuts

 time limit per test: 4 seconds
 memory limit per test: 256 megabytes

Having the magical beanstalk, Jack has been gathering a lot of peanuts lately. Eventually, he has obtained n pockets of peanuts, conveniently numbered 1 to n from left to right. The i -th pocket has a_i peanuts.

Jack and his childhood friend Alice decide to play a game around the peanuts. First, Alice divides the pockets into some boxes; each box will have a non-zero number of **consecutive** pockets, and each pocket will, obviously, belong to exactly one box. At the same time, Alice does not change the order of the boxes, that is, the boxes are numbered in ascending order of the indices of the pockets in them.

After that, Alice and Jack will take turns alternately, with Alice going first.

At each turn, the current player will remove a positive number of peanuts from **exactly one** pocket which belongs to the **leftmost non-empty box** (i.e., the leftmost box containing at least one non-empty pocket). In other words, if we number the boxes from left to right, then each player can only pick peanuts from the pocket in the j -th box ($j \geq 2$) only if the $(j - 1)$ -th box has no peanuts left. The player who cannot make a valid move loses.

Alice is sure she will win since she has the advantage of dividing the pockets into boxes herself. Thus, she wanted to know how many ways there are for her to divide the peanuts into boxes at the start of the game so that she will win, assuming both players play optimally. Can you help her with the calculation?

As the result can be very large, output it modulo 998 244 353.

Input

Each test consists of multiple test cases. The first line contains a single integer t ($1 \leq t \leq 10^4$) — the number of test cases. The description of the test cases follows.

The first line of each test case contains a single integer n ($1 \leq n \leq 10^6$) — the number of pockets.

The second line of each test case contains n integers a_1, a_2, \dots, a_n ($1 \leq a_i \leq 10^9$) — the number of peanuts in each pocket.

It is guaranteed that the sum of n over all test cases does not exceed 10^6 .

Output

For each test case, output a single integer — the number of ways for Alice to divide the pockets into boxes at the start of the game to guarantee her win, assuming both players play optimally, modulo 998 244 353.

Example

input

Copy

```
5
3
1 2 3
4
1 2 3 1
5
1 1 1 1 1
2
1 1
10
```

Codeforces Round 983 (Div. 2)

Contest is running

00:04:47

Contestant



→ Submit?

 Language: Python 3.8.10
 Almost always, if you send a solution on PyPy, it works much faster

 Choose file: Choose File No file chosen

Be careful: there is 50 points penalty for submission which fails the pretests or resubmission (except failure on the first test, denial of judgement or similar verdicts). "Passed pretests" submission verdict doesn't guarantee that the solution is absolutely correct and it will pass system tests.

Submit

→ Last submissions

Submission	Time	Verdict
289276953	Nov/01/2024 19:21	Wrong answer on pretest 1

→ Score table

	Score
Problem A	274
Problem B	411
Problem C	685
Problem D	959
Problem E	1233
Problem F	1644
Successful hack	100
Unsuccessful hack	-50
Unsuccessful submission	-50
Resubmission	-50

* If you solve problem on 01:53 from the first attempt

1 2 3 4 5 6 7 8 9 10

output

Copy

```
1
4
16
0
205
```

Note

In the first test case, the only way for Alice to win is to divide the pockets into two boxes as follows: $([1, 2], [3])$ (the first box contains the first two pockets and the second box contains the third pocket). Alice wins by taking both peanuts from the second pocket, leaving Jack with $([1], [3])$. Jack is forced to take the only peanut left in the first box, which allows Alice to take the remaining ones in the second box.

In the second test case, the winning divisions for Alice are $([1], [2, 3, 1])$, $([1, 2, 3, 1])$, $([1, 2], [3], [1])$, and $([1, 2], [3, 1])$.

In the third test case, Alice always wins no matter how she divides the pockets into boxes.

In the fourth test case, Alice always loses no matter how she divides the pockets into boxes.

[Codeforces](#) (c) Copyright 2010-2024 Mike Mirzayanov

The only programming contests Web 2.0 platform

Server time: Nov/01/2024 22:28:08^{UTC+6} (j3).

Desktop version, switch to [mobile version](#).

[Privacy Policy](#)

Supported by

**ITMO**