Problem 9: Gates to Infinite Winnings :money_mouth: 3+6 Points

Problem ID: `gates`
Rank: `3+3`

Introduction

Due to a recent surge in customers, the CALICasinO has struggled to accommodate all guests. Despite attempts to increase capacity by adding new games such as a pinball machine and Genshin Impact, there was still an increasingly long queue of guests who were still waiting for their daily dose of dopamine. This was until a quant intern came up with a brilliant strategy: having people gamble to get in! However, they are indecisive on what game to use, and have offered a bounty to any customer who can provide a good solution.

In order to win back some losses from the pinball machine, Big Ben has submitted a game. In the scheme, there are N lightbulbs which work like bits, and there is a slot machine to determine the gates in between the bulbs. If the equation with the lightbulbs and gates turns out to be 1, the door to the CALICasino will open.

To add a twist to the game, Big Ben has also added two more slots at the end. These work as parentheses; the first slot rolls a number p from 0 to $N-1$ determining the position of the left parenthesis, and the second slot rolls q from p to $N-1$ determining the position of the right parenthesis. These parentheses work as in normal math; the value inside the parentheses is computed first before the rest of the expression.

Given N and a series of lightbulbs, find the number of distinct ways to open the gate.

© 2025 CALICO Informatics Competition

Problem Statement

We have $\mathbf{N}$ bits (0 or 1) $\mathbf{A}_0...\mathbf{A}_{N-1}$. We want to put $N-1$ gates (AND, OR, XOR) g_i between these bits such that the equation

$$\mathbf{A}_0 \; g_1 \; \mathbf{A}_1 \; g_2 \; \mathbf{A}_2 \; \cdots \; g_{N-2} \; \mathbf{A}_{N-2} \; g_{N-1} \; \mathbf{A}_{N-1} = 1.$$

Additionally, we must add **one left parenthesis** before some bit A_i, and **one right parenthesis** after any bit A_j, such that $i <= j$.

A configuration is **distinct** if the gates are different or the parentheses are at different locations. Find the total number of distinct gate configurations that satisfy the equation.

All expressions are evaluated strictly from left to right. This means, for example, that without any parentheses, `1 | 0 & 0 = (1 | 0) & 0 = 0`.

Input Format

The first line of the input contains a single integer $\mathbf{T}$ denoting the number of test cases that follow. For each test case:

- The first line contains a single integer $\mathbf{N}$, denoting the number of interview problems.
- The next line contains N space-separated integers $\mathbf{A}_0, \ldots, \mathbf{A}_{\mathbf{N}-1}$. All $\mathbf{A}_i$ are either 0 or 1.

Output Format

For each test case, print an integer denoting the number of ways to satisfy the equation in the problem statement by placing gates and parentheses, modulo **1000000007**.

Constraints

$1 \leq \mathbf{T} \leq 100$

Main Test Set

$1 \leq \mathbf{N} \leq 10^3$

It is guaranteed the sum of $\mathbf{N}$ across all test cases is less than 10^3.

Bonus Test Set

$1 \leq \mathbf{N} \leq 2 * 10^5$

It is guaranteed the sum of $\mathbf{N}$ across all test cases is less than $2 \cdot 10^5$.

Sample Test Cases

Main Sample Input Download

```
2
2
1 1
3
1 0 1
```

Main Sample Output Download

```
6
36
```

Main Sample Explanations

For test case #1, AND and OR gates will work. Parentheses do not affect the value, but can still be placed in three positions: $(0,0),(0,1),(1,1)$. Therefore, we have $2 \cdot 3 = 6$ ways.

For test case #2, there are 36 possible ways to satisfy the equation.

© 2025 CALICO Informatics Competition