# Counting Permutations

| | |
|---|---|
| Input file: | `standard input` |
| Output file: | `standard output` |
| Time limit: | 1 second |
| Memory limit: | 512 megabytes |

You are given an array $a$ of size $n$ and two integers $m_1$ and $m_2$. We call array $p$ *good* if the following conditions are met:

- $p$ is a permutation of size $n$, consisting of different integers from 1 to $n$.
- The array $a_{p_1} \bmod m_1,\ a_{p_2} \bmod m_1,\ \ldots,\ a_{p_n} \bmod m_1$ is a non-decreasing array.
- The array $a_{p_1} \bmod m_2,\ a_{p_2} \bmod m_2,\ \ldots,\ a_{p_n} \bmod m_2$ is a non-increasing array.

Count the number of good permutations $p$ by modulo 998244353.

## Input

The input consists of multiple test cases. The first line contains a single integer $t$ ($1 \le t \le 10\,000$) — the number of test cases. The description of the test cases follows.

The first line of each test case contains three integers $n$, $m_1$ and $m_2$ ($1 \le n \le 100\,000$, $1 \le m_1, m_2 \le 10^4$) — the length of the array and two modules.

The next line contains $n$ integers $a_1, a_2, \ldots, a_n$ ($1 \le a_i \le 10^9$) — the array.

It is guaranteed that the sum of $n$ for all test cases does not exceed 100 000.

## Output

For each test case, print a single integer — the answer to the problem by modulo 998244353.

## Example

| standard input | standard output |
|---|---|
| 3<br>5 2 3<br>1 2 3 4 10<br>4 2 4<br>1 2 3 4<br>3 8 9<br>1 1 1 | 2<br>0<br>6 |

## Note

In the first test case, there are two good permutations $p = [2, 4, 5, 1, 3]$ and $p = [2, 5, 4, 1, 3]$.

In the second test case, there are no good permutations.

In the third test case, all 6 permutations are good.