

Problem G. Permutation

Time limit 1500 ms

Memory limit 512MB

Problem Description

You are an archivist in the Grand Library of the Prime Order. The library holds a dazzling sequence of n ancient runes, each rune distinct, numbered from 1 to n , and you must arrange them perfectly before the midnight bells toll. However, the runes are mischievous: whenever a rune with a larger symbol appears before a smaller one, it creates an **inversion**. Likewise, whenever a rune is immediately followed by a smaller one, it forms a **descent**.

Formally, given a permutation a_1, a_2, \dots, a_n ,

- An **inversion** is a pair of indices (i, j) such that $1 \leq i < j \leq n$ and $a_i > a_j$.
- A **descent** is an index i such that $1 \leq i < n$ and $a_i > a_{i+1}$.

For given integers of n, k, x , count in how many ways you can order these n distinct runes so that exactly k inversions occur, and exactly x descents occur. Since the answer might be very large, output it modulo 998244353.

Input format

The first line contains one integer t ($1 \leq t \leq 5 \times 10^5$) - the number of test cases.

Then following t lines, each containing three integers n, k, x ($1 \leq n \leq 1000, 1 \leq k, x \leq 20$).

Output format

For each test case, output one integer in one line - the answer to the problem modulo 998244353.

Subtask score

Subtask	Score	Additional Constraints
0	0	Sample testcases
1	3	$n \leq 10$
2	41	$1 \leq k, x \leq 8$
3	56	No additional constraints

Sample

Sample Input 1

```
4
3 1 1
10 5 3
123 8 6
1000 20 20
```

Sample Output 1

2

527

977494316

274895549