6909 Kevin's Problem

In his applied probability class, Kevin learned about the Secretary Problem.

> There are N applicants applying for secretary position in a company. The applicants are interviewed by a hiring manager one by one in arbitrary order. During the interview, the manager can rank the applicant's quality uniquely relative to all applicants which have been interviewed so far, but he has no idea of the quality of applicants yet to be interviewed. The decision whether the manager should hire or reject the applicant must be done right after the interview ended (before he starts an interview with other applicant) and cannot be changed, i.e. once rejected, that particular applicant cannot be considered anymore. There is only one position available, so what is the best decision strategy for this problem?

One reasonable strategy is: reject the first K applicants and hire the first remaining applicant who is better than all previous interviewed applicants, or hire the last one if there is no such applicant.

Unfortunately, Kevin did not pay a full attention in his class. He misunderstood the strategy; instead of "...hire the first remaining applicant who is better than all previous interviewed applicants ...", he thought it is "...hire the first remaining applicant who is better than the (immediate) previous interviewed applicant ...". Let's call this variation as Kevin's strategy.

Given N, K, and p determine in how many ways (interview order) such that the applicant whose rank is p among all applicants will be selected by Kevin's strategy. For example, let $N = 4$, $K = 2$, and $p = 2$. Among all permutation of $1 \ldots N$, there are only 7 possible permutation such that the 2nd rank applicant is selected by Kevin's strategy.

- 1, 3, 2, 4 — 2 is selected because it's better than 3.
- 1, 3, 4, 2 — 2 is selected because 4 is worse than 3, and 2 is better than 4.
- 1, 4, 2, 3
- 3, 1, 4, 2
- 3, 4, 2, 1
- 4, 1, 3, 2
- 4, 3, 2, 1

Note that the first 2 applicants will not be hired in this example ($K = 2$). Clearly, the 2nd rank applicant will not be selected in any permutation where she appears in the first K.

An applicant has a better rank if her rank is higher (smaller number), e.g. 2nd rank is better than 5th.

Input

The first line of input contains an integer T ($T \leq 100$) denoting the number of cases. Each case contains three integers: N, K, and p ($2 \leq N \leq 500$; $1 \leq K < N$; $1 \leq p \leq N$) as explained in the problem description above.

Output

For each case, output ‘`Case #`X`:` Y’, where X is the case number starts from 1 and Y is the number of permutation of $1 \ldots N$ such that the p-th rank applicants is selected by Kevin's strategy for that particular case. As this number could be very large, *modulo* Y by 1,000,000,007.

Explanation for 3rd sample case:

There are 5 applicants and Kevin's strategy will reject the first 3. 5th rank applicant will be selected only if she is interviewed last, thus the manager has no choice but to hire her. Among 24 possible permutations where 5th rank applicant appears last, only 12 permutations which make her hired:

```
1, 2, 3, 4, 5       2, 1, 3, 4, 5       3, 1, 2, 4, 5       4, 1, 2, 3, 5
1, 3, 2, 4, 5       2, 3, 1, 4, 5       3, 2, 1, 4, 5       4, 2, 1, 3, 5
1, 4, 2, 3, 5       2, 4, 1, 3, 5       3, 4, 1, 2, 5       4, 3, 1, 2, 5
```

Some examples where 5th rank will not be hired even though she is the last:

`1, 2, 4, 3, 5` – 3rd rank will be hired.
`1, 4, 3, 2, 5` – 2nd rank will be hired.
`3, 2, 4, 1, 5` – 1st rank will be hired.
...

Sample Input

```
4
4 2 2
5 2 3
5 3 5
8 4 2
```

Sample Output

```
Case #1: 7
Case #2: 26
Case #3: 12
Case #4: 7890
```