

Finding Maximum

One way of finding the maximum element in an array is to initialize a variable to the first element in the array, iterate through the remaining array, and update the variable whenever a value strictly greater than it is found. Assuming that the array contains N elements each in the range $1..K$, how many such arrays exist such that the above algorithm performs exactly P updates? Initialization of the variable is not counted as an update.

For example, the possible arrays for $N = 4$, $K = 3$ and $P = 2$ are:

1. {1, 1, 2, 3}
2. {1, 2, 1, 3}
3. {1, 2, 2, 3}
4. {1, 2, 3, 1}
5. {1, 2, 3, 2}
6. {1, 2, 3, 3}

Input

The first line contains T the number of test cases. There follow T lines, containing 3 space separated integers N , K and P .

Output

Output T lines, one for each test case. On each line, output the answer as asked above. Since the answers can get very big, output the answer modulo 1000000007.

Example

Sample Input:

```
3
4 3 2
2 3 1
3 4 1
```

Sample Output:

```
6
3
30
```

Constraints

$1 \leq T \leq 100$
 $1 \leq n \leq 100$
 $1 \leq K \leq 300$
 $0 \leq P < n$