

Festival de música

MUSICA

En un festival de música se van a reproducir n canciones en orden aleatorio. Algunas canciones animan más al público que otras; en concreto, se sabe que el público le da una nota a_i a la i -ésima canción.



Decimos que se produce un *subidón* cuando se reproduce una canción que es mejor que la anterior según el público; es decir, si las canciones se reproducen siguiendo una permutación de índices p (donde p es una permutación de $\{1, 2, \dots, n\}$), decimos que la posición i -ésima (para $2 \leq i \leq n$) es un subidón si $a_{p_i} > a_{p_{i-1}}$. Decimos que una reproducción de canciones tiene k subidones si existen exactamente k posiciones que son un subidón.

Dado k , cuenta cuántas permutaciones p existen tales que, si se reproducen las canciones siguiendo la permutación p (primero a_{p_1} , después a_{p_2} , etc.), hay exactamente k subidones. Como este número puede ser muy grande, da la respuesta módulo $10^9 + 7$.

Entrada y salida

La primera línea de la entrada contiene el número de casos T .

Por cada caso, habrá una línea de entrada con los enteros n y k .

La siguiente línea de cada caso contiene n enteros a_i , las notas del público para las n canciones.

Para cada caso, debes imprimir un entero: el número de permutaciones, módulo $10^9 + 7$, que producen exactamente k subidones.

Restricciones

- $1 \leq T \leq 2000$
- $1 \leq n \leq 2000$
- $0 \leq k \leq n - 1$
- La suma de n para todos los casos es como mucho 2000.
- $1 \leq a_i \leq 10^9$

Subtareas

1. (8 puntos) La suma de n para todos los casos es como mucho 9.
2. (21 puntos) $k = 0$.
3. (32 puntos) a es una permutación de $\{1, 2, \dots, n\}$.
4. (39 puntos) Sin restricciones adicionales.

Ejemplo

Entrada	Salida
3	4
3 1	4
1 2 3	0
4 2	
3 3 2 2	
5 4	
3 3 3 3 3	

En el **primer caso**, tenemos 3 canciones con notas 1, 2 y 3 y queremos conseguir exactamente 1 subidón. Existen 6 permutaciones posibles de los índices. Si analizamos las secuencias de notas que generan:

- Si reproducimos las canciones en el orden de índices [1, 2, 3], las notas son 1, 2, 3. Hay 2 subidones (de 1 a 2, y de 2 a 3).
- Si el orden es [3, 2, 1], las notas son 3, 2, 1. Hay 0 subidones.
- Los 4 órdenes restantes generan exactamente 1 subidón. Por ejemplo, el orden de índices [1, 3, 2] genera las notas 1, 3, 2, donde solo el salto de 1 a 3 es un subidón. Las otras válidas son [2, 1, 3], [2, 3, 1] y [3, 1, 2]. Por lo tanto, la respuesta es 4.

En el **segundo caso**, tenemos 4 canciones con notas 3, 3, 2, 2 y queremos exactamente 2 subidones. Para conseguir 2 subidones con estas notas, la única estructura válida para la secuencia reproducida es alternar las notas bajas y altas: 2, 3, 2, 3. Como permutar los índices importa, las 4 permutaciones de índices válidas que generan esta secuencia de notas son: [3, 1, 4, 2], [3, 2, 4, 1], [4, 1, 3, 2] y [4, 2, 3, 1].

En el **tercer caso**, tenemos 5 canciones y todas tienen la misma nota (3). Queremos conseguir 4 subidones. Como la condición para un subidón es que la nota de la canción sea *estrictamente mayor* que la anterior ($a_{p_i} > a_{p_{i-1}}$), es imposible conseguir un subidón si todas las notas son iguales. El número de subidones siempre será 0, por lo que existen 0 permutaciones que cumplan lo que se pide.