Arranging the Ornaments

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	256 megabytes

The New Year tree had not yet been set up in the university hall, so Diana stretched a ribbon across the hall and hung N ornaments of different sizes along it. If the ornaments are numbered from 1 to N in increasing order of size, their numbers from left to right form a permutation $A = (A_1, \ldots, A_N)$.

— If only they were arranged from smallest to largest, — Karina sighed.

Alexander called for volunteers, but only K people responded. They decided to divide the ribbon into at most K nonempty consecutive segments, assigning each segment to one volunteer. A volunteer removes all ornaments from their segment, places them in a box, cleans them, and hangs them back on the same segment, this time in increasing order of size from left to right. All ornaments must be cleaned, so every ornament must belong to some segment. A box can hold at most L ornaments; therefore, every segment must contain between 1 and L ornaments.

— The result depends on how we divide the ribbon, — Ilshat noted. — I wonder how many different arrangements we can get?

Help the organizers determine how many distinct arrangements of ornaments on the ribbon can be obtained in this way. Identical arrangements obtained from different partitions of the ribbon are counted only once. If it is impossible to partition the ornaments according to these rules, output 0.

Since the answer may be very large, output it modulo 998244353.

Input

The first line contains an integer t ($1 \le t \le 50$) — the number of test cases.

The first line of each test case contains three integers N, K, and L ($1 \le N \le 300$, $1 \le K, L \le N$). The second line contains N pairwise distinct integers $A_1, \ldots, A_N$ from 1 to N.

The sum of N over all test cases in one input file does not exceed 600. Some subtasks in the scoring table impose stricter limits on this sum.

Output

For each test case, output on a separate line the number of distinct resulting sequences modulo 998244353.

Scoring

The tests are divided into subtasks listed in the table below. Points for a subtask are awarded only if the solution successfully passes all tests in that subtask and all subtasks listed in the column «Required subtasks».

Let S — the sum of N over all test cases in one input file. The restriction on S applies to the entire file, while the other restrictions in the table are satisfied for each test case.

Subtask	Points	Additional constraints		Required subtasks
		S	Other conditions	
0	0	Examples from the statement		—
1	10	$S \leq 600$	$L = 1$	—
2	10	$S \leq 300$	$K = 1$	—
3	12	$S \leq 600$	$A_i = i$ for all i	—
4	13	$S \leq 600$	$A_i = N + 1 - i$ for all i	—
5	15	$S \leq 300$	$K \leq 2$	2
6	10	$S \leq 32$	$N \leq 16$	—
7	7	$S \leq 600$	$K = N$	—
8	7	$S \leq 600$	$L = N$	—
9	6	$S \leq 200$	$N \leq 100$	6
10	5	$S \leq 300$	$K \leq 10$	2, 5
11	5	$S \leq 600$	No additional constraints.	0–10

Example

standard input	standard output
4 4 3 3 2 1 4 3 3 1 2 3 2 1 5 3 2 1 2 3 4 5 1 1 1 1	4 0 1 1

Note

In the first test case, the following arrangements can be obtained: $(1, 2, 3, 4)$, $(1, 2, 4, 3)$, $(2, 1, 3, 4)$, and $(2, 1, 4, 3)$.

In the second test case, a single segment of length at most 2 cannot cover all three elements. In the third test case, valid partitions exist, and each of them produces $(1, 2, 3, 4, 5)$.