Array Modify

Input file:	`standard input`
Output file:	`standard output`
Time limit:	3 seconds
Memory limit:	256 megabytes

Given an array A with n numbers and two integers L, m. You have to output the final array after performing the following operation m times:

$$A[i] = \left(\sum_{j=i}^{\min(i+L-1,n)} A[j] \right) \bmod P, \text{ in order of } i = 1, 2, \cdots, n$$

$P = 998,244,353$ is a fixed number.

Input

The first line of input file contains an integer T ($1 \leq T \leq 20$), describing the number of test cases.

Then there are $2 \times T$ lines, with every two lines representing a test case.

The first line of each test case contains three integers: n, L, m ($1 \leq n \leq 100,000$, $1 \leq L \leq n$, $1 \leq m \leq 10^9$) described above.

The second line of that contains exactly n integers,the i-th of which represents $A[i]$ ($0 \leq A[i] < P$, for $i = 1, 2, \cdots, n$).

It is guaranteed that the sum of n in all cases does not exceed $200,000$.

Output

You should output exactly T lines.

For each case, print **Case d:** (d represents the order of test case) first, and then output exactly n integers, i^{th} of which represents $A[i]$, separated by exactly one space.

Example

standard input	standard output
2 3 2 2 1 2 3 4 1 3 1 2 3 4	Case 1: 8 8 3 Case 2: 1 2 3 4

Note

Sample 1:

	Calculation	A
Original	-	$1, 2, 3$
First Operation	$A[1] = (A[1] + A[2]) \bmod P = (1 + 2) \bmod P = 3$	$3, 2, 3$
	$A[2] = (A[2] + A[3]) \bmod P = (2 + 3) \bmod P = 5$	$3, 5, 3$
	$A[3] = (A[3]) \bmod P = 3$	$3, 5, 3$
Second Operation	$A[1] = (A[1] + A[2]) \bmod P = (3 + 5) \bmod P = 8$	$8, 5, 3$
	$A[2] = (A[2] + A[3]) \bmod P = (5 + 3) \bmod P = 8$	$8, 8, 3$
	$A[3] = (A[3]) \bmod P = 3$	$8, 8, 3$