Summing Sums

The N (1 ≤ N ≤ 50,000) cows, conveniently numbered 1, 2, ..., N, are trying to learn some encryption algorithms. After studying a few examples, they have decided to make one of their own! However, they are not very experienced at this, so their algorithm is very simple:

Each cow i is given a starting number C_i (0 ≤ C_i < 90,000,000), and then all the cows perform the following process in parallel:

- First, each cow finds the sum of the numbers of the other N-1 cows.
- After all cows are finished, each cow replaces her number with the sum she computed. To avoid very large numbers, the cows will keep track of their numbers modulo 98,765,431.

They told Canmuu the moose about it in November; he was quite impressed. Then one foggy Christmas Eve, Canmuu came to say:

> "Your algorithm is too easy to break! You should repeat it T (1 ≤ T ≤ 1,414,213,562) times instead."

Obviously, the cows were very frustrated with having to perform so many repetitions of the same boring algorithm, so after many hours of arguing, Canmuu and the cows reached a compromise: You are to calculate the numbers after the encryption is performed!

Input

- Line 1: Two space-separated integers: N and T.
- Lines 2..N+1: Line i+1 contains a single integer: C_i.

Output

- Lines 1..N: Line i contains a single integer representing the number of cow i (modulo 98,765,431) at the end of the encryption.

Example

Input:
```
3 4
1
0
4
```

Output:
```
26
25
29
```

The following is a table of the cows' numbers for each turn:

	Cows' numbers		
Turn	Cow1	Cow2	Cow3
0	1	0	4
1	4	5	1

```
2      6   5   9
3     14  15  11
4     26  25  29
```

Warning: large input/output data.