Problem D
All Kill
Time Limit: ? Second(s)

You just finished participating in a programming contest with your friend. Unfortunately, you were unable to All Kill the contest (i.e., solve all of the problems), but you are now wondering if there might be some strategy that would have solved all of the problems.

Solving a problem has two phases, a thinking phase and a coding phase. Your friend is responsible for all the thinking while you are responsible for all the coding.

For each problem, you've computed exactly how long it would take for you to code. However, before you can code a problem in contest, your friend needs to get the idea to solve it first. You aren't sure how to estimate the time when your friend gets a solution idea, so you model it like this: For every problem, your friend gets the idea of how to solve this problem at a uniformly random minute during the contest. Each of these is an independent random variable. You can only code one problem at a time, so there may be several problems queued up at any moment of time. You always prioritize coding the lowest numbered problems first. You do this minute-by-minute, so you will switch to coding a lower-numbered problem if your friend gets the idea for it before you're finished coding a higher-numbered problem, but you would prefer not to do this. Context switching is an expensive operation, even in the human brain!

The contest strategy can be modeled as follows for each minute:

- For each problem that doesn't yet have an idea, your friend will get the idea to solve it with probability 1/(number of minutes remaining in contest). Your friend can get the idea to solve multiple problems in the same minute.
- Among the problems that still need code time and your friend has gotten the solution idea, you will take the lowest numbered one and spend the next minute coding it (if no problem satisfies the condition, you do nothing at this step).

You would like to know the probability of these two events happening together:

- Your team finishes coding all the problems by the end of the contest
- For each problem, the time spent coding that problem is a contiguous interval

Let p be this probability, n be the number of problems in the contest and t be the number of minutes in the contest. It can be shown that $p \cdot t^n$ is an integer. Output the value of $(p \cdot t^n) \pmod{998{,}244{,}353}$. Note that $998{,}244{,}353$ is a large prime.

Input

The first line of input contains two space-separated integers n ($1 \leq n \leq 10^5$) and t ($1 \leq t \leq 10^8$), where there were n problems in the contest, and the contest lasted t minutes.

Each of the next n lines contains a single integer x ($1 \leq x \leq 10^3$). These are the times to code each problem in minutes, in order. It is guaranteed that the sum of these times is less than or equal to t.

Output

Output a single integer, which is $(p \cdot t^n) \pmod{998{,}244{,}353}$, where p is the probability of the two events mentioned above happening together.

Sample Input 1	Sample Output 1
3 5 1 2 1	60

Sample Input 2	Sample Output 2
5 5 1 1 1 1 1	1296