

# Go go Baron Bunny!

Input file:            standard input  
Output file:           standard output  
Time limit:            1 second  
Memory limit:         256 megabytes



Bad Frog is not willing to pay very much for programming competitions, and would provide only  $n$  brain cells to memory programming competition knowledge points.

In the morning, Bad Frog's teammates find that there are  $k$  knowledge points in Bad Frog's mind and they cost  $a_1, a_2, \dots, a_k$  brain cells respectively.

Since Bad Frog hardly practices, the number of brain cells to keep each knowledge point will decrease by 1 every day, and when the number becomes 0, Bad Frog will forget the knowledge point.

Every night, teammates find that Bad Frog does no practice and the brain cells to memory knowledge points decreases, so they will push Bad Frog to use all the remaining brain cells to learn a new knowledge point.

For example, if Bad Frog provides 22 brain cells and costs 1, 4, 7, 1, 5, 4 brain cells to keep knowledge points respectively in the morning, then at night the brain cells cost will become 0, 3, 6, 0, 4, 3 respectively, and then Bad Frog will forget the 2 knowledge points that are kept with zero brain cells. Finally, under the pressure from teammates, Bad Frog will learn a new knowledge point and use all the 6 remain brain cells to keep it so that the brain cells cost will become 3, 6, 4, 3, 6 respectively.

Denote  $A = \{a_1, a_2, \dots, a_k\}$  as the multiset of the number of brain cells to keep knowledge points. Now there are  $t$  days to go before the competition. In order to have a stable performance, teammates hope that the **multiset**  $A$  on the competition day ( $t$  days from now) to be the same as it is today. Determine the number of **arrays**  $(a_1, a_2, \dots, a_k)$  satisfying this condition, modulo 998244353.

## Input

$n k t$

Constraints:

- $1 \leq k \leq n \leq 10^{12}$
- $1 \leq t \leq 10^{12}$

## Output

Print the answer modulo 998244353, as an integer.

## Examples

standard input	standard output
2 1 2	1
8 4 2	6
29 7 154	0
50 10 10	77225400

## Note

- For the first case, the only one array is  $[2]$  ( $[2] \rightarrow [1, 1] \rightarrow [2]$ ).
- For the second case, the 6 arrays are  $[1, 1, 3, 3]$ ,  $[1, 3, 1, 3]$ ,  $[1, 3, 3, 1]$ ,  $[3, 1, 1, 3]$ ,  $[3, 1, 3, 1]$ ,  $[3, 3, 1, 1]$ .