Almost Video Game III (videogame2)

In `Video Game II: Almost Video Game III`, you control a character which has K abilities, numbered from 0 to $K-1$. The objective of this game is to defeat N bosses, which are numbered from 0 to $N-1$, in the order you have to beat them.

Figure 1: A screenshot from the videogame.

At the beginning of a playthrough, none of the K abilities has been researched yet. Before facing each boss, you can research a (possibly empty) subset of the K abilities. Once an ability is researched, it will stay available for the rest of the playthrough and you do not have to research it again.

After beating all bosses, the *score* of the playthrough is equal to b^2, where b is the number of bosses where your character had all K abilities researched.

Let's consider the following playthrough for $N = 4$ and $K = 3$:

- Before fighting boss 0, you research ability 2.
- You fight boss 0 with 1 ability researched.
- Before fighting boss 1, you don't research any new abilities.
- You fight boss 1 with 1 ability (ability 2) researched.
- Before fighting boss 2, you research abilities 0 and 1.
- You fight boss 2 with all $K = 3$ abilities researched.
- Before fighting boss 3, you don't research any new abilities.
- You fight boss 3 with all $K = 3$ abilities researched.

Since you fought bosses 3 and 4 with all $K = 3$ abilities researched, b is equal to 2 and the score of this playthrough is $b^2 = 4$.

What is the sum of the scores of all possible playthroughs? Two playthroughs are considered different if there exists a boss i ($0 \leq i < N$) that you fight having a different subset of abilities researched.

Since the answer can be large, print its remainder modulo MOD, where MOD is a prime number given in the input.

☞ Among the attachments of this task you may find a template file `videogame2.*` with a sample incomplete implementation.

Input

The first line of input contains three integers N, K and MOD: the number of bosses, the number of abilities, and the modulo.

Output

Print one integer, the sum of the scores of all possible playthroughs modulo MOD.

✍ The *modulo* operation (a mod m) can be written in C/C++/Python as `(a % m)` and in Pascal as `(a mod m)`. To avoid the *integer overflow* error, remember to reduce all partial results through the modulus, and not just the final result!
Notice that if $x < 10^9 + 7$*, then* $2 \cdot x$ *fits into a* C/C++ `int` *and* Pascal `longint`.

Constraints

- $1 \leq N \leq 10^{18}$.
- $1 \leq K \leq 1\,000\,000$.
- $10^8 \leq MOD \leq 10^9 + 7$ and MOD is a prime number.

Scoring

Your program will be tested against several test cases grouped in subtasks. In order to obtain the score of a subtask, your program needs to correctly solve all of its test cases.

– **Subtask 1** (0 points) Examples.

– **Subtask 2** (9 points) $N, K \leq 100$.

– **Subtask 3** (5 points) $N, K \leq 1500$.

– **Subtask 4** (8 points) $N \leq 1\,000\,000$.

– **Subtask 5** (7 points) $K \leq 2$.

– **Subtask 6** (15 points) $K \leq 5$.

– **Subtask 7** (29 points) $K \leq 100$.

– **Subtask 8** (18 points) $K \leq 1500$.

– **Subtask 9** (9 points) No additional constraints.

Examples

input	output
3 2 100000007	26
5 3 998244353	517
999013 97 998244853	116848898
958613246711292682 1000000 1000000007	112173097

Explanation

In the **first sample case**, there are $N = 3$ bosses and $K = 2$ abilities.

A playthrough is denoted as a sequence of $N = 3$ sets $s_0 \subseteq s_1 \subseteq s_2$, where:

- $s_0 \subseteq \{0, 1\}$ is the subset of the abilities researched before facing boss 0;
- $s_1 \subseteq \{0, 1\}$ is the subset of the abilities researched before facing boss 1;
- $s_2 \subseteq \{0, 1\}$ is the subset of the abilities researched before facing boss 2.

In total, there are 16 different playthroughs:

- $[\{\}, \{\}, \{\}]$, with a score of $0^2 = 0$.
- $[\{\}, \{\}, \{0\}]$, with a score of $0^2 = 0$.
- $[\{\}, \{0\}, \{0\}]$, with a score of $0^2 = 0$.
- $[\{0\}, \{0\}, \{0\}]$, with a score of $0^2 = 0$.
- $[\{\}, \{\}, \{1\}]$, with a score of $0^2 = 0$.
- $[\{\}, \{1\}, \{1\}]$, with a score of $0^2 = 0$.
- $[\{1\}, \{1\}, \{1\}]$, with a score of $0^2 = 0$.
- $[\{\}, \{\}, \{0, 1\}]$, with a score of $1^2 = 1$.
- $[\{\}, \{0\}, \{0, 1\}]$, with a score of $1^2 = 1$.
- $[\{0\}, \{0\}, \{0, 1\}]$, with a score of $1^2 = 1$.
- $[\{\}, \{1\}, \{0, 1\}]$, with a score of $1^2 = 1$.
- $[\{1\}, \{1\}, \{0, 1\}]$, with a score of $1^2 = 1$.
- $[\{\}, \{0, 1\}, \{0, 1\}]$, with a score of $2^2 = 4$.
- $[\{0\}, \{0, 1\}, \{0, 1\}]$, with a score of $2^2 = 4$.
- $[\{1\}, \{0, 1\}, \{0, 1\}]$, with a score of $2^2 = 4$.
- $[\{0, 1\}, \{0, 1\}, \{0, 1\}]$, with a score of $3^2 = 9$.

The sum of the scores across all possible playthroughs is $0 \cdot 7 + 1 \cdot 5 + 3 \cdot 4 + 9 = 0 + 5 + 12 + 9 = 26$.