Bacteria Growth

Ethan collects many different kinds of bacteria. Each type of bacteria is kept in its own cage. Ethan is only interested in bacteria that reproduce by splitting into k copies of themselves (including the original). For example, say Ethan has a single bacteria that splits into 3 copies of itself each day. Then at the end of the first day, he has 3 bacteria, at the end of the second day, he has 9 bacteria, at the end of the third day, he has 27 bacteria, and at the end of the fourth day he has 81 bacteria!

Ethan's family is going on a trip to Disney world for d days. Given the splitting factor (k) for each bacteria, can you tell him how many of each kind of bacteria he'll have waiting for him when he comes back?

Since the answer may be large, print each value modulo $10^9 + 7$.

Grading

Correctness & Efficiency: 80%

- Passes 40 test cases: 80%
- Passes 30 to 39 test cases: 60%
- Passes 20 to 29 test cases: 40%
- Passes 1 to 19 test cases: 20%
- Passes 0 test cases: 0%

Code Quality: 20%

Input Format

Each test case consists of a single line with three space separated integers, n, k and d. n is the initial amount of that species of bacteria Ethan has, k is the amount it splits by each day, and d is the number of day's Ethan's family will be gone.

Constraints

$1 \leq n, k \leq 10^9$

$1 \leq d \leq 10^{18}$

Output Format

For each test case, print a single line of output containing the number of bacteria that will be there when Ethan gets back. Since your answer may be large, print it modulo $10^9 + 7$.

Sample Input 0

```
1 3 4
```

Sample Output 0

```
81
```

Explanation 0

Matches the example in the problem statement.

Sample Input 1

```
2 4 1
```

Sample Output 1

```
8
```

Sample Input 2

```
3 6 2
```

Sample Output 2

```
108
```

Explanation 2

Day 0: 3 Day 1: 18 Day 2: 108

Sample Input 3

```
1000000000 1000000000 1000000000
```

Sample Output 3

```
812102106
```

```cpp
#include <cmath>
#include <cstdio>
#include <vector>
#include <iostream>
#include <algorithm>
using namespace std;


int main() {
    /* Enter your code here. Read input from STDIN. Print output to STDOUT */
    return 0;
}

```

Line: 1 Col: 1

Upload Code as File Test against custom input Run Code Submit Code