Moon Safari (medium)

Air is a music duo from France.
You will be told the secret of the critically acclaimed album Moon Safari: mathematics.
The goal of your new task is to compute an ethereal sum.

$$\sum_{i=1}^{N} a^i i^r$$

Three trips on the moon are provided, Moon (easy), Moon1 (medium), Moon2 (hard) with different constraints.

Input

The first line contains an integer *T*, the number of test cases.
On the next *T* lines, you will be given three integers *N*, *a* and *r*.

Output

Output *T* lines, one for each test case, with $S_{N,a,r}$ = sum(*a^i i^r*, for *i* in [*1..N*]).

Since the answer can get very big, output it modulo 10^9+7.

Example

Input:
2
3 4 5
6 7 8

Output:
16068
329990641

Explanation

The first case is, with *N*=3, *a*=4, *r*=5, about the sum : 4^1 × 1^5 + 4^2 × 2^5 + 4^3 × 3^5 = 4 + 512 + 15552 = **16068**.
The second case is, with *N*=6, *a*=7, *r*=8, about the sum : 7^1 × 1^8 + 7^2 × 2^8 + 7^3 × 3^8 + 7^4 × 4^8 + 7^5 × 5^8 + 7^6 × 6^8 + 7^7 × 7^8 = 204329992069 ≡ **329990641** (mod 10^9+7).

Constraints

1 < T
1 < r
1 < N < 10^9
1 < a < 10^9
(T < 1000 and r < 18) or (T < 100 and r < 72) or (T < 10 and r < 256) or (T = 1 and r < 444)

Information

This trip can be done with a O(T×r^2×log(N)) method and some interpreted languages.
My MOON1-Py3 code got AC in 9.00s for the 4 input files.
(My MOON2 code got AC in 0.00s with C, 0.18s with Py2.7, 0.35 with Py3.2)
Good luck and have fun ;-)