Project Euler #219: Skew-cost coding

This problem is a programming version of Problem 219 from projecteuler.net

Let A and B be bit strings (sequences of 0 and 1).
If A is equal to the leftmost $length(A)$ bits of B, then A is said to be a prefix of B.
For example, 00110 is a prefix of 001101001, but not of 00111 or 100110.

A prefix-free code of size n is a collection of n distinct bit strings such that no string is a prefix of any other. For example, this is a prefix-free code of size 6:

- 0000, 0001, 001, 01, 10, 11

Now suppose that it costs one penny to transmit a 0 bit, but 4 pence to transmit a 1.
Then the total cost of the prefix-free code shown above is 35 pence, which happens to be the cheapest possible for the skewed pricing scheme in question.
In short, we write $Cost(6, 1, 4) = 35$.

Given several tuples of numbers (n, a, b) find the total cost of the cheapest prefix-free code of size n with costs a and b of transmission 0 bit and 1 bit respectively.

Calculate the result modulo 1000000007 ($10^9 + 7$).

Input Format

First line of each test file contains a single integer q that is the number of tuples. Then q lines follow, each containing three integers: n, a and b — size of prefix-free code, cost of 0 and cost of 1.

Constraints

- $1 \leq q \leq 100$
- $2 \leq n \leq 10^{16}$
- $1 \leq a \leq 10$
- $1 \leq b \leq 10$

Output Format

Print exactly q lines with a single integer on each: an answer to the corresponding query modulo $10^9 + 7$
.

Sample Input 0

```
2
6 1 4
9 1 1
```

Sample Output 0

```
35
29
```

Explanation 0

The first prefix-free code is the following:
0000, 0001, 001, 01, 10, 11
Its cost is $4 + 7 + 6 + 5 + 5 + 8 = 35$

The second prefix-free code is the following:
000, 001, 010, 011, 100, 101, 110, 1110, 1111
Its cost is $3 + 3 + 3 + 3 + 3 + 3 + 3 + 4 + 4 = 29$. This code is not unique.