DMOPC '20 Contest 3 P3 - A Ring of Buckets

Time limit: 2.0s **Memory limit:** 256M

Lily has a ring of N buckets, numbered from 1 to N. Each bucket has capacity M. She has a pouring bucket with capacity $K+2$, and wants to fill all buckets completely without any overflow, that is, to M **exactly**. Unfortunately, every time she tries to pour into a bucket, she spills a little, and 1 unit is spilled into each adjacent bucket, with K poured into her original bucket. Note that 1 and N are adjacent.

Lily wants to fill all of the buckets to capacity, but wants to have zero overflow. Being a genius, Lily already knows the answer, and challenges you to find it too.

However, Lily will quickly bore of you listing out the N numbers, so she decides on the following formula. If A_i is the number of times you must pour into the ith bucket, Lily will choose an arbitrary number B and ask you to compute $A_1 + A_2 \times B + A_3 \times B^2 + \cdots + A_N \times B^{N-1}$. This number may be large, so Lily will be satisfied if you can output it modulo 10^9+7. Additionally, if there are multiple solutions, choose the one that has the lexicographically smallest value of $A_1, A_2, \ldots, A_N$.

Finally, Lily has a lot of buckets, so she will ask you Q questions, each with their own values of N, M, K and B. Can you answer them all?

Constraints

For all subtasks:

$1 \leq M, K \leq 10^9$

$2 \leq B \leq 10^9$

$3 \leq N \leq 10^9$

$1 \leq Q \leq 10^4$

Subtask 1 [1/15]

$Q = 1$

$N, M, K \leq 5$

Subtask 2 [2/15]

$Q = 1$

$N, M, K \leq 400$

Subtask 3 [3/15]

$Q = 1$

$K = 1$

$1 \leq N, M \leq 10^6$

Subtask 4 [3/15]

$Q = 1$

$K = 2$

$1 \leq N, M \leq 10^6$

Subtask 5 [3/15]

$Q = 1$

$3 \leq K \leq 10^6$

$1 \leq N, M \leq 10^6$

Subtask 6 [3/15]

No additional constraints.

Input Specification

The first line will contain Q, the number of questions.

The next Q lines will contain four integers each, N, M, K, B.

Output Specification

Output Q lines. For each question, if it is impossible to fill all the buckets exactly to capacity, output `-1`.

Otherwise, output the required integer as specified above. Don't forget to output it modulo $10^9 + 7$.

Sample Input 1

```
1
4 4 1 100
```

Sample Output 1

```
-1
```

Explanation for Sample Output 1

There is no way to fill all the buckets exactly.

Sample Input 2

```
1
3 4 2 7
```

Sample Output 2

```
57
```

Explanation for Sample Output 2

If we pour once into each bucket, we get a solution array `1 1 1`. Then, our required value is:

$$1 + 1 \times 7 + 1 \times 7^2 = 57$$

Sample Input 3

```
1
999999 999999 5 8
```

Sample Output 3

```
35952588
```