Problem B. Multi-Ladders

Input file: standard input
Output file: standard output
Time limit: 4 seconds
Memory limit: 1024 megabytes

In order to attract customers in a store, the owner decided to order a special neon sign to be placed in front of the store. This special neon sign is composed by several ladder-type components. Each ladder-type component can be represented by a ladder graph L_n that is a planar, undirected graph with $2n$ vertices and $3n-2$ edges. Each vertex represents a light bulb of the neon sign, and each edge represents a wire connecting two light bulbs. The ladder graph can be obtained as the Cartesian product of two path graphs, one of which has only one edge: $L_n = P_n \times P_2$, where P_n is a path of n vertices. Figure 1 shows L_6. The main frame for holding k ladder-type components is k-regular polygon. Each edge of k-regular

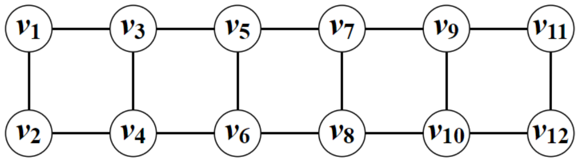

Figure 1: L_6.

polygon is combined with the top edge of L_n. Figure 2 shows the case with $k = 3$ and $n = 4$, and Figure 3 shows the case with $k = 4$ and $n = 3$. For ease of description, the neon sign is represented by a graph $G = (V, E)$ comprised by the bulbs (vertices) and wires (edges). To make the neon lights more dazzling,

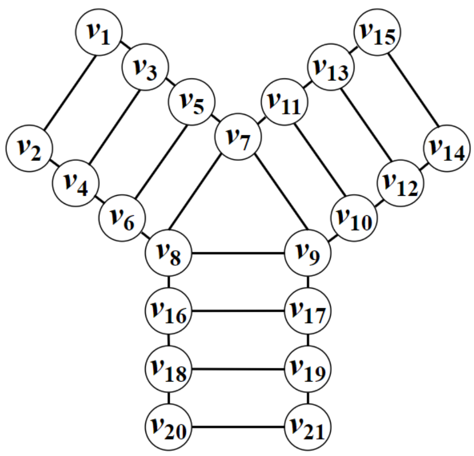

Figure 2: An example of $k = 3$ and $n = 4$.

the designer, Ray, came up with a way to specify the color of the bulbs (vertices) in G. Ray would like to assign colors to blubs such that the following condition hold. A coloring of G is said to be *proper* if we assign colors to the vertices of G so that if u and v are adjacent, then the colors assigned to u and v are different. The bulbs in $G = (V, E)$ are distinguished, that is, the bulbs are named by labels $v_1, v_2, \dots v_m$. Consequently, two color assignments of bulbs in G will be considered different if a proper coloring of the bulbs of G that uses at most λ colors is a function f, with domain V and codomain $\{1, 2, 3, \dots, \lambda\}$, where $f(u) \neq f(v)$, for adjacent vertices $u, v \in V$. Proper colorings are then different in the same way that these functions are different. The maximum number of different ways to color G using λ colors is called the *critical number* of G.

Given (1) n (for L_n), (2) k (for k-regular polygon), and (3) the number of available colors λ, your task is to compute the critical number of G. Note that if the result is larger than or equal to $10^9 + 7$, you

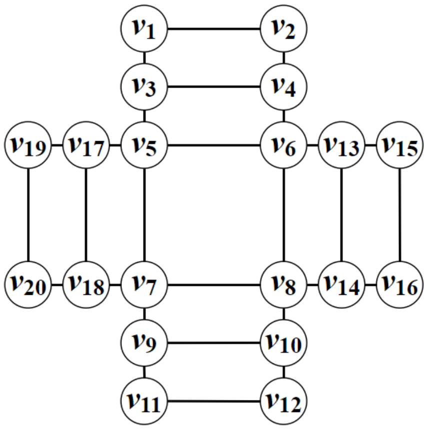

Figure 3: An example of $k = 4$ and $n = 3$.

should output the value modulo $10^9 + 7$, that is, the remainder obtained using the actual value divided by $10^9 + 7$.

Input

The first line of the input file contains an integer L ($L \leq 20$) that indicates the number of test cases as follows. For each test case, the first line contains three integers (separated by whitespaces) representing n, k, and λ.

Constraints

- $1 \leq n \leq 1000000000$.
- $3 \leq k \leq 1000000000$ for each test case.
- $0 \leq \lambda \leq 1000000000$.

Output

The output contains one line for each test case. Each line contains one non-negative integer representing the critical number of G. Note that if the result is larger than or equal to $10^9 + 7$, you should output the value modulo $10^9 + 7$, that is, the remainder obtained using the actual value divided by $10^9 + 7$.

Examples

standard input	standard output
1 2 3 3	162