10444 Multi-Peg Towers of Hanoi

In 3-Peg Tower of Hanoi problem n disks, all of unequal radiuses are placed initially on a peg 1 with smaller disks on top of larger disks. Using empty peg 2 the whole tower should be shifted to peg 3 with the condition that only one disk at a time from top of any peg can be moved to the top of another peg. Never a larger disk can be placed on top of a smaller one. While shifting the tower in minimum moves requires a simple recursive strategy of shifting $n-1$ disks to the only intermediate peg, number of moves required is exponential. That is if there are n disks to move then $2^n - 1$ moves are necessary. If there are $p > 3$ pegs then the task becomes much easier but now finding the optimal strategy is more difficult. In fact nobody has ever been able to prove that a certain strategy is optimal. Researchers, however, have found a strategy that many believe to be optimal, although could not prove so. This strategy is known to be *Presumed Optimal Solution* (POS). In POS the strategy is the following.

In shifting a tower of n disks in p-peg system a certain optimal number n_1 disks are shifted to an intermediate peg P_i. Then remaining $n - n_1$ disks are shifted to destination using a total of $p-1$ pegs, since peg P_i is loaded with smaller disks, and therefore, cannot be used for transfer of larger disks. This strategy is employed recursively for solving any sub-problem. While such a strategy can result in different POS solutions the number of moves remains the same.

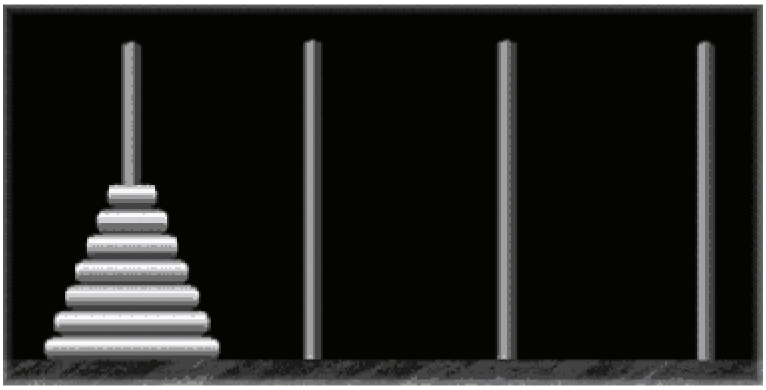

Fig: A four-peg tower of Hanoi

It is known that in POS solutions each disk requires 2^k moves to reach destination for some integer k. Furthermore, maximum number of disks each of which requires 2^k moves to reach destination is equal to ${}^{p-3+k}C_k$ (number of combinations of $(p-3+k)$ things taken k at a time), provided that such a number of disks is available. Moreover, in a POS strategy it is possible to transfer disks greedily. That is, if there are disks each of which requires 2^{k+1} moves to reach destination, then number of disks each of which require 2^k moves to reach destination is ${}^{p-3+k}C_k$ (number of combinations of $(p-3+k)$ things taken k at a time). Given n, p you will have to determine the minimum number of moves required to move the n disks from source to destination using p pegs and the process described above.

Input

The input file contains several lines of input. Each line contains three integers n $(0 \le n \le 200)$, p $(3 < p \le 20)$. The meanings of these two integers are explained in the problem statement. Input is terminated by a line containing two zeroes. This line should not be processed.

Output

For each line of input (that are asked to be processed) produce one line of output. This line should contain the serial of the output followed by an integer N that indicates the number of movements required to move all the disks from source peg to destination peg.

Sample Input

```
3 4
4 4
10 4
10 5
0 0
```

Sample Output

```
Case 1: 5
Case 2: 9
Case 3: 49
Case 4: 31
```