

A: Deck of Cards

Suppose we start with a deck of n unique cards and deal the entire deck out to k players in the usual way: the top card is given to player 1, the next is given to player 2, the k^{th} card is given to player k , the $k+1^{\text{st}}$ to player 1, and so on. Next, we will pick up the cards, by placing player 1's cards on top, followed by player 2's cards, and so on, until player k 's cards are on the bottom. Also, each player's cards are in reverse order – the last card they were dealt is on top, and the first card they were dealt is on the bottom.

How many times, including the first, must this process be repeated before the deck returns to its original order?

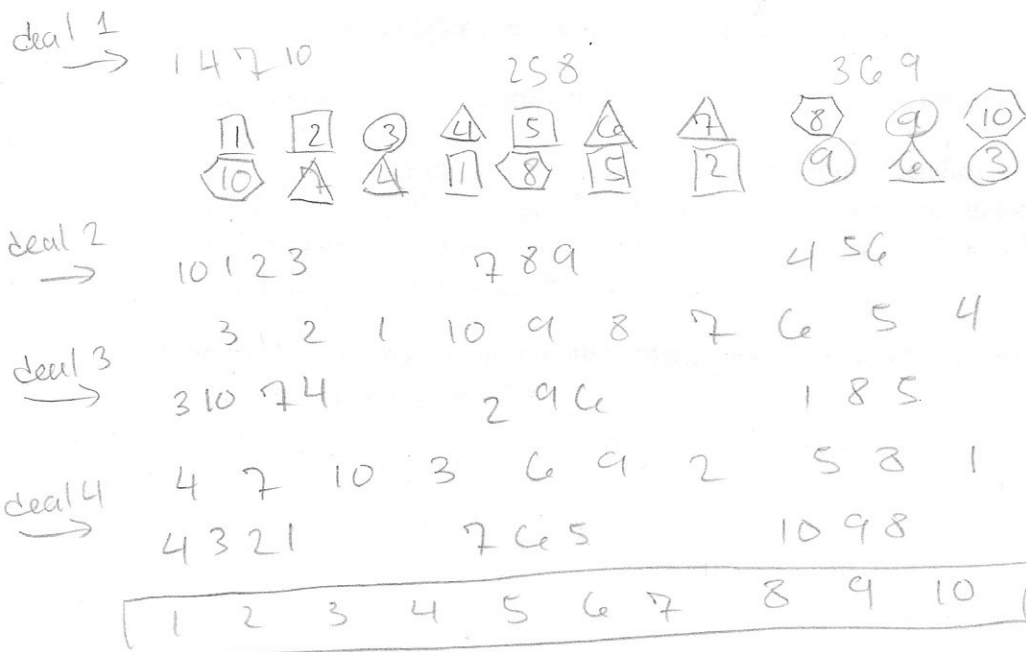
The Input

The input file will contain a maximum of 100 test cases. Each test case will consist of a single line with two integers, n and k ($1 \leq n \leq 800$, $1 \leq k \leq 800$). The input will end with a line containing two 0's.

The Output

For each test case in the input, print a single integer, indicating the number of deals required to return the deck to its original order. Output each integer on its own line, with no extra spaces, and no blank lines between answers. All possible inputs yield answers which will fit in a signed 64-bit integer.

| n k Sample Input | Sample Output |
|-------------------------|---------------|
| 1 3 | 1 |
| 10 3 | 4 |
| 52 4 | 13 |
| 0 0 | |



$$b_k + a$$

$$b = n/k = 10/3 = 3$$

$$a = n \% k = 10 \% 3 = 1$$

$$3 \cdot 3 + 1 = 10$$