We define super digit of an integer x using the following rules:

Given an integer, we need to find the super digit of the integer.

- If x has only 1 digit, then its super digit is x.
- Otherwise, the super digit of x is equal to the super digit of the sum of the digits of x.

For example, the super digit of 9875 will be calculated as:

```
super_digit(9875)       9+8+7+5 = 29
super_digit(29)         2 + 9 = 11
super_digit(11)         1 + 1 = 2
super_digit(2)          = 2
```

Example

$n =' 9875'$

$k = 4$

The number p is created by concatenating the string n k times so the initial $p = 9875987598759875$.

```
superDigit(p) = superDigit(9875987598759875)
              9+8+7+5+9+8+7+5+9+8+7+5+9+8+7+5 = 116
superDigit(p) = superDigit(116)
              1+1+6 = 8
superDigit(p) = superDigit(8)
```

All of the digits of p sum to 116. The digits of 116 sum to 8. 8 is only one digit, so it is the super digit.

Function Description

Complete the function superDigit in the editor below. It must return the calculated super digit as an integer.

superDigit has the following parameter(s):

- string n: a string representation of an integer
- int k: the times to concatenate n to make p

Returns

- int: the super digit of n repeated k times

Input Format

The first line contains two space separated integers, n and k.

Constraints

- $1 \leq n < 10^{100000}$
- $1 \leq k \leq 10^5$

Change Theme Language Python 3

```python
#!/bin/python3

import math
import os
import random
import re
import sys

#
# Complete the 'superDigit' function below.
#
# The function is expected to return an INTEGER.
# The function accepts following parameters:
#  1. STRING n
#  2. INTEGER k
#
def superDigit(n, k):
    # Write your code here
    num_sum = 0
    for char in n:
        num_sum += int(char)

    # (n)k string
    num_sum = num_sum * k

    return helper(str(num_sum))


def helper(num_string):
    if len(num_string) == 1:
        return int(num_string)

    num_sum = 0
    for char in num_string:
        num_sum += int(char)

    return helper(str(num_sum))
```

Line: 68 Col: 1

Upload Code as File Test against custom input Run Code Submit Code

Congratulations

You solved this challenge. Would you like to challenge your friends?

Test case 0
Test case 1
Test case 2
Test case 3
Test case 4
Test case 5

Compiler Message

```
Success
```

Input (stdin) Download

```
148 3
```

Expected Output Download

```
3
```