Recursive Digit Sum

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
```

You are given two numbers n and k. The number p is created by concatenating the string n k times. Continuing the above example where $n = 9875$, assume your value $k = 4$. Your initial $p = 9875\ 9875\ 9875\ 9875$ (spaces added for clarity).

```
superDigit(p) = superDigit(9875987598759875)
              5+7+8+9+5+7+8+9+5+7+8+9+5+7+8+9 = 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's the super digit.

Function Description

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

Author wanbo

Difficulty Medium

Max Score 30

Submitted By 19322

NEED HELP?

View discussions

View editorial

View top submissions

RATE THIS CHALLENGE

MORE DETAILS

Download problem statement

Download sample test cases

Suggest Edits

superDigit has the following parameter(s):

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

Input Format

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

Constraints

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

Output Format

Return the super digit of p, where p is created as described above.

Sample Input 0

```
148 3
```

Sample Output 0

```
3
```

Explanation 0

Here $n = 148$ and $k = 3$, so $P = 148148148$.

```
super_digit(P) = super_digit(148148148)
               = super_digit(1+4+8+1+4+8+1+4+8)
               = super_digit(39)
               = super_digit(3+9)
               = super_digit(12)
               = super_digit(1+2)
               = super_digit(3)
               = 3.
```

Sample Input 1

```
9875 4
```

Sample Output 1

```
8
```

Sample Input 2

```
123 3
```

Sample Output 2

```
9
```

Explanation 2

Here $n = 123$ and $k = 3$, so $P = 123123123$.

```
super_digit(P) = super_digit(123123123)
               = super_digit(1+2+3+1+2+3+1+2+3)
               = super_digit(18)
               = super_digit(1+8)
               = super_digit(9)
               = 9
```

Java 8

```
1   import java.io.*;
2   import java.math.*;
3   import java.security.*;
4   import java.text.*;
5   import java.util.*;
6   import java.util.concurrent.*;
7   import java.util.regex.*;
8
9   public class Solution {
10
11      public static BigInteger sum_n_big(String n) {
12          BigInteger sd = BigInteger.valueOf(0);
13          for (int i = 0; i < n.length(); i++) {
14              sd = sd.add(
15                  BigInteger.valueOf(
16                      Character.getNumericValue(n.charAt(i))
17                  )
18              );
19          }
20          return sd;
21      }
```

Line: 34 Col: 53

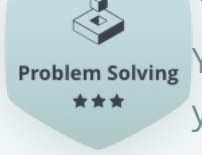
Upload Code as File

Test against custom input

Run Code

Submit Code

You have earned 30.00 points!



You have earned 30.00 points!

You are now 234 points away from the 4th star for your problem solving badge.

15% 241/475

Congratulations

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

Next Challenge

Testcase 0 | Testcase 1 | Testcase 2 | Testcase 3 | **12 Testcases**

Input (stdin) Download

```
148  3
```

Expected Output Download

```
3
```

Compiler Message

```
Success
```