Codewriting **300**

Consider a sequence of numbers a_0, a_1, ..., a_n, in which an element is equal to the sum of squared digits of the previous element. The sequence ends once an element that has already been in the sequence appears again.

Given the first element `a0`, find the length of the sequence.

Example

- For `a0 = 16`, the output should be `solution(a0) = 9`.

  Here's how elements of the sequence are constructed:

  - $a_0 = 16$
  - $a_1 = 1^2 + 6^2 = 37$
  - $a_2 = 3^2 + 7^2 = 58$
  - $a_3 = 5^2 + 8^2 = 89$
  - $a_4 = 8^2 + 9^2 = 145$
  - $a_5 = 1^2 + 4^2 + 5^2 = 42$
  - $a_6 = 4^2 + 2^2 = 20$
  - $a_7 = 2^2 + 0^2 = 4$
  - $a_8 = 4^2 = 16$, which has already occurred before (a_0)

  Thus, there are `9` elements in the sequence.

- For `a0 = 103`, the output should be `solution(a0) = 4`.

  The sequence goes as follows: `103 -> 10 -> 1 -> 1`, `4` elements altogether.

Input/Output

- **[execution time limit] 4 seconds (py3)**
- **[input] integer a0**

  First element of a sequence, positive integer.

  Guaranteed constraints:
  $1 \le a0 \le 10^5$.

- **[output] integer**

[Python 3] Syntax Tips

```
# Prints help message to the console
# Returns a string
def helloWorld(name):
    print("This prints to the console when you Run Tests")
    return "Hello, " + name
```