Codewriting **300**

We want to turn the given integer into a number that has only one non-zero digit using a tail rounding approach. This means that at each step we take the last non `0` digit of the number and round it to `0` or to `10`. If it's less than `5` we round it to `0` if it's larger than or equal to `5` we round it to `10` (rounding to `10` means increasing the next significant digit by `1`). The process stops immediately once there is only one non-zero digit left.

Example

- For `n = 15`, the output should be `solution(n) = 20`;
- For `n = 1234`, the output should be `solution(n) = 1000`.

  `1234 -> 1230 -> 1200 -> 1000`.
- For `n = 1445`, the output should be `solution(n) = 2000`.

  `1445 -> 1450 -> 1500 -> 2000`.

Input/Output

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

  A positive integer.

  Guaranteed constraints:
  $1 \leq value \leq 10^8$.
- **[output] integer**

  The rounded number.

[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
```