Square digit chains

Problem 92 (http://projecteuler.net/problem=92)

A number chain is created by continuously adding the square of the digits in a number to form a new number until it has been seen before.

For example,

44 → 32 → 13 → 10 → 1 → 1 85 → 89 → 145 → 42 → 20 → 4 → 16 → 37 → 58 → 89

Therefore any chain that arrives at 1 or 89 will become stuck in an endless loop. What is most amazing is that EVERY starting number will eventually arrive at 1 or 89.

How many starting numbers below ten million will arrive at 89?

Solution

In [1]:
```
def qd(n):
    s = 0
    while n:
        r = n % 10
        s += r * r
        n = n // 10
    return s

def test(n):
    while n != 1 and n != 89:
        n = qd(n)
    return n

sum(test(n) == 89 for n in range(1, 10_000_000))
```

Out[1]: 8581146