

206.隐藏的平方数(Concealed Square)

找到唯一一个正整数，它的平方具有 1_2_3_4_5_6_7_8_9_0 的形式，其中 “_” 表示一位数。

分析：满足题目中要求形式的平方数最小的是 1020304050607080900，也就是所有下划线代表的数字都是零；而最大的平方数应该是 1929394959697989990，也就是所有下划线代表的数字都是九。分别取这两个数的平方根，也就是我们想要找的数 N 的上界和下界，有：

$$1010101010 \leq N \leq 1389026623$$

此外，满足题目要求的平方数最后一位数是 0，则其平方根的最后位数也应该是 0，因此我们可以从 1389026620 开始每次递减 10，求这个数的平方，看它满不满足题目要求。找到满足要求的数后返回即可。代码如下：

```
# time cost = 1.58 ms ± 17.4 μs
```

```
def is_concealed_square(number):
    seq = ['1', '2', '3', '4', '5', '6', '7', '8', '9', '0']
    digits = [x for x in str(number)]
    if len(digits) == 19 and digits[0::2] == seq:
        return True
    return False

def main(n=1389026620):
    while True:
        if is_concealed_square(n**2):
            return n
        else:
            n = n - 10
```