**69. Sqrt(x)**

Easy

Implement int sqrt(int x).

Compute and return the square root of *x*, where *x* is guaranteed to be a non-negative integer.

Since the return type is an integer, the decimal digits are truncated and only the integer part of the result is returned.

**Example 1:**

**Input:** 4

**Output:** 2

**Example 2:**

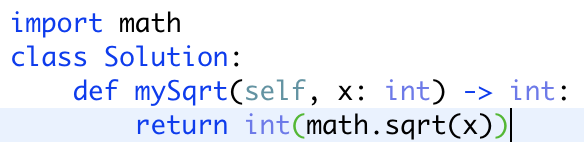
**Input:** 8

**Output:** 2

**Explanation:** The square root of 8 is 2.82842..., and since

  the decimal part is truncated, 2 is returned.

注意是舍弃小数，不是四舍五入



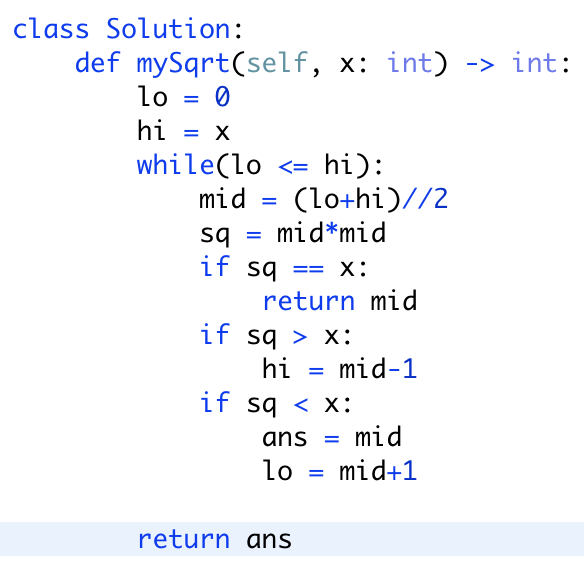
这么写应该会被打吧

1. 二分法

因为只需要找整数部分，所以可以用二分搜索去找哪一个数的平方是x。

注意：

1. 对于非平方数（即找不到一个平方刚好等于x的数），因为开方后小数部分被省略，如取整为2，肯定是<8的，所以需要保留记录每次小于x情况下的数。当二分搜索最后找不到一个平方刚好等于x的数时，可以return，就是其整数部分。
2. Testcase为0的情况



1. 牛顿迭代法

求平方根：，可以转为求这个方程的根。 然后就可以使用牛顿迭代法，详细文献参考<https://matongxue.com/madocs/205.html>

简单来说就是求曲线的根点（与x轴相交的点，即y=0时的解）可以用方程来迭代求解，从而不断逼近根点。这里, 然后带入上面公式可得。 这里n是题目里x，xk是函数定义域上任意值（驻点除外）。对于此题xk初始值应该为x。

