**Finding the Square Root of an Integer**

Find the square root of the integer without using any Python library. You have to find the floor value of the square root.

For example if the given number is 16, then the answer would be 4.

If the given number is 27, the answer would be 5 because sqrt(5) = 5.196 whose floor value is 5.

The expected time complexity is O(log(n))

Here is some boilerplate code and test cases to start with:

**def** **sqrt**(number):

"""

Calculate the floored square root of a number

Args:

number(int): Number to find the floored squared root

Returns:

int: Floored Square Root

"""

**pass**

**print** ("Pass" **if** (3 == sqrt(9)) **else** "Fail")

**print** ("Pass" **if** (0 == sqrt(0)) **else** "Fail")

**print** ("Pass" **if** (4 == sqrt(16)) **else** "Fail")

**print** ("Pass" **if** (1 == sqrt(1)) **else** "Fail")

**print** ("Pass" **if** (5 == sqrt(27)) **else** "Fail")