Q.1] Given a non-negative integer x, return the square root of x rounded down to the nearest integer. The returned integer should be non-negative as well. You must not use any built-in exponent function or operator.

**Example 1:**

**Input:** x = 4 **Output:** 2 **Explanation:** The square root of 4 is 2, so we return 2.

**Example 2:**

**Input:** x = 8 **Output:** 2 **Explanation:** The square root of 8 is 2.82842..., and since we round it down to the nearest integer, 2 is returned.

**Constraints:**

0 <= x <= 2^31 - 1

**Note:**Create a GitHub file for the solution and add the file link the the answer section below.

Solution :

class Solution:

def mySqrt(self, x: int) -> int:

left = 0

right = x

while left <= right:

mid = (left + right) // 2

if mid \* mid > x:

right = mid - 1

else:

left = mid + 1

return right