69. Sqrt(x)

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.

- For example, do not use `pow(x, 0.5)` in c++ or `x ** 0.5` in python.

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
```

Constraints:

- $0 <= x <= 2^{31} - 1$

Java

```java
class Solution {
    public int mySqrt(int x) {
long start=1,end=x,ans=0;
while(start<=end){
    long mid=(start+end)/2;

    if(mid*mid==x){
    return (int)mid;
        }
        else if(mid*mid<x){
      start = mid + 1;
        ans=mid;
            }
            else{
                end = mid - 1;

                }

    }
    return (int)ans;
    }
}
```

Notes

Custom Testcase (Contribute)

Run Submit

Submission Result: Accepted (/submissions/detail/827808872/)

More Details > (/submissions/detail/827808872/)

Next challenges: Pow(x, n) (/problems/powx-n)

Share your acceptance!

16

Check out our solution! Reveal Solution > (/articles/sqrtx/)

Notes