

## Leetcode Problem 1. (Easy)

### 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 nearest integer, 2 is returned.

#### Constraints:

- $0 \leq x \leq 2^{31} - 1$

Link: <https://leetcode.com/problems/sqrtx/>

```
class Solution {
    public int mySqrt(int x) {

        if (x == 0 || x == 1) {
            return x;
        }

        int left = 1;
        int right = x;
        int result = 0;

        while (left <= right) {
            int mid = left + (right - left) / 2;

            if (mid <= x / mid) {
```

```

        left = mid + 1;
        result = mid;
    } else {
        right = mid - 1;
    }
}

return result;
}
}

```

Problem List
Premium

Description

Editorial

Solutions (6.5K)

Submissions

Accepted

Next question

70. Climbing Stairs

More challenges

50. Pow(x, n)

367. Valid Perfect Square

All statuses

All languages

Accepted

a few seconds ago

Java

Sakib Rahman

Apr 19, 2023 11:32

Java

Runtime 1 ms

Beats 100%

Memory 39.8 MB

Beats 74.33%

Notes

Write your notes here

Related Tags

Select tags

0/5

```

class Solution {
    public int mySqrt(int x) {
        if (x == 0 || x == 1) {
            return x;
        }

        int left = 1;
        int right = x;
        int result = 0;

        while (left <= right) {
            int mid = left + (right - left) / 2;

```

Console

Run

Submit