

Square Root of a Number [LeetCode](#)

Given a positive integer x , find its square root with specified precision.

Approach 1: Brute force approach to find the square root of a number

The function `sqrtBruteForce` implements a brute force approach to find the square root of x .

It iterates from 1 to x and checks the square of each number. If the square matches x , it returns the number as the square root.

If the square is less than x , it updates the answer to the current number. If the square exceeds x , it breaks the loop.

The time complexity of this approach is $O(x)$, and the space complexity is $O(1)$.

Approach 2: Binary search approach to find the square root of a number

The function `sqrt` implements a binary search approach to find the square root of x .

It initializes low as 0 and high as x to define the search space.

It iteratively calculates the middle number (`mid`) using the binary search technique.

It checks the square of `mid` and compares it with x . If they are equal, `mid` is returned as the square root.

If the square is greater than x , the search is narrowed down to the lower half by updating high. If the square is less than x , the search is narrowed down to the upper half by updating low.

The time complexity of this approach is $O(\log(x))$, and the space complexity is $O(1)$.

Function to find the precision part of the square root of number.

The function `findPrecision` adjusts the precision of the square root found using one of the above approaches.

It takes the integer square root (`intAns`), the input number x , and the desired precision as input.

It iteratively adjusts the precision by dividing the factor by 10 at each iteration.

It then performs a loop to find the square root with the updated precision by incrementing `ans` with factor until the square exceeds or equals x .

The time complexity of this function depends on the precision specified. If the precision is a constant, the time complexity is $O(\text{precision})$, where precision is the number of decimal places desired. The space complexity is $O(1)$.