

Solution approach explanation for the Prime Number Problem:

Traditionally, the solution for finding out whether the input number is prime or not can be achieved by finding the number of factors the given number is having. Here, we can measure the count of numbers which are dividing the given number iterating from 2 to the given number. The number having a factor count as exact 2 will be a prime factor. In such approach, we are using a for loop and this makes the complexity of the problem as $O(n)$.

So, to increase the speed of the algorithm by decreasing the complexity, I followed a better approach. Here, as we know, smallest prime factor of a composite number N is less than or equal to \sqrt{N}

Hence, in the function, I used the iterations from 2 to \sqrt{N} to find any available factors. If we find any such factors, the function returns False, concluding that the given number is not a prime number. The function returns True otherwise. The time complexity for this solution approach is $O(\sqrt{n})$, which is more efficient.