Problem 762: Prime Number of Set Bits in Binary Representation

Problem Information

Difficulty: Easy
Acceptance Rate: 72.02%
Paid Only: No
Tags: Math, Bit Manipulation

Problem Description

Given two integers `left` and `right`, return _the**count** of numbers in the **inclusive** range _`[left, right]`_having a**prime number of set bits** in their binary representation_.

Recall that the **number of set bits** an integer has is the number of `1`'s present when written in binary.

* For example, `21` written in binary is `10101`, which has `3` set bits.

Example 1:

Input: left = 6, right = 10 **Output:** 4 **Explanation:** 6 -> 110 (2 set bits, 2 is prime) 7 -> 111 (3 set bits, 3 is prime) 8 -> 1000 (1 set bit, 1 is not prime) 9 -> 1001 (2 set bits, 2 is prime) 10 -> 1010 (2 set bits, 2 is prime) 4 numbers have a prime number of set bits.

Example 2:

Input: left = 10, right = 15 **Output:** 5 **Explanation:** 10 -> 1010 (2 set bits, 2 is prime) 11 -> 1011 (3 set bits, 3 is prime) 12 -> 1100 (2 set bits, 2 is prime) 13 -> 1101 (3 set bits, 3 is prime) 14 -> 1110 (3 set bits, 3 is prime) 15 -> 1111 (4 set bits, 4 is not prime) 5 numbers have a prime number of set bits.

Constraints:

* `1 <= left <= right <= 106` * `0 <= right - left <= 104`

Code Snippets

C++:

```
class Solution {
public:
int countPrimeSetBits(int left, int right) {

}
};
```

Java:

```
class Solution {
public int countPrimeSetBits(int left, int right) {

}
}
```

Python3:

```
class Solution:
def countPrimeSetBits(self, left: int, right: int) -> int:
```