Special Prime 2

locked

Problem | Submissions | Leaderboard | Discussions

Submissions: 33
Max Score: 10
Difficulty: Medium

Rate This Challenge:

More

While inspecting prime numbers, you find that some prime numbers are "special".

A special prime number is one which is the sum of two smaller **consecutive primes** and 1.

Example: 31 = 13 + 17 + 1 (notice that 13 and 17 are **consecutive primes**).

You want to look for other special prime numbers, so you inspect primes in the range 2 to 1000 inclusive.

You want to inspect different subranges, so you use n and inspect the range 2 to n inclusive.

Given n and k, find whether or not there are at least k special primes in the range 2 to n inclusive.

Output 1 if there are at least k special primes in the range, or 0 otherwise.

Use brute force for your solution.

Input Format

- One line containing N and K

Constraints

- N is between 2 and 1000 inclusive
- k is between 0 and 1000 inclusive

Output Format

One line containing 1 if there are at least k special primes in the range, or 0 otherwise.

Sample Input 0

```
26 2
```

Sample Output 0

```
1
```

Explanation 0

Two special primes between 2 and 26 inclusive are:

19 = 7 + 11 + 1

and

13 = 5 + 7 + 1.

Sample Input 1

```
45 6
```

Sample Output 1

```
0
```

Explanation 1

In the range 2 to 45, there is less than 6 special primes.

C++20

```cpp
#include <cmath>
#include <cstdio>
#include <vector>
#include <iostream>
#include <algorithm>
using namespace std;


int main() {
    /* Enter your code here. Read input from STDIN. Print output to STDOUT */
    return 0;
}
```

Line: 1 Col: 1

Upload Code as File | Test against custom input | Run Code | Submit Code