The statement of this exercise is identical to that of exercise . But here the solution required is more efficient in general.

Given a natural number \( n \), let \( s(n) \) be the sum of the digits of \( n \). In this exercise, we say that \( n \) is a perfect prime if the infinite sequence \( n, s(n), s(s(n)), \ldots \) only contains prime numbers. For instance, 977 is a perfect prime, because 977, \( 9 + 7 + 7 = 23 \), \( 2 + 3 = 5 \), \( 5 \), \ldots, are all prime numbers.

Write a recursive function that tells if a natural number \( n \) is a perfect prime or not.

**Interface**

- C++: `bool isPerfectPrime(int n);`
- C: `int isPerfectPrime(int n);`
- Java: `public static boolean isPerfectPrime(int n);`
- Python: `isPerfectPrime(n) # returns bool
  isPerfectPrime(n: int) -> bool`

**Precondition**

We have \( n \geq 0 \).

**Observation**

You only need to submit the required procedure; your main program will be ignored.

**Problem information**

Author: Salvador Roura
Translator: Carlos Molina
Generation: 2023-07-14 18:01:08

https://jutge.org