Jutge.org

The Virtual Learning Environment for Computer Programming

Extended Fibonacci numbers — P25832_en

Catorzè Concurs de Programació de la UPC - Final (2016-09-21)

The well known Fibonacci numbers are defined recursively as follows: $F_0 = 0$, $F_1 = 1$, $F_i = F_{i-1} + F_{i-2}$ for $i \geq 2$. The first Fibonacci numbers are 0, 1, 1, 2, 3, 5, 8, 13, 21, $\ldots$.

Let us generalize the Fibonacci numbers. For every pair of natural numbers a and b, define the sequence $S(a,b) = [f_0, f_1, \ldots]$ as $f_0 = a$, $f_1 = b$, $f_i = f_{i-1} + f_{i-2}$ for $i \geq 2$. Note that $S(0,1)$ is the traditional Fibonacci sequence.

You are given a natural number n. Please compute how many pairs (a,b) exist such that $S(a,b)$ has a $i \geq 3$ where $f_i = n$. For instance, for $n = 2$ there are exactly three such sequences: $S(0,1) = [0,1,1,2,\ldots]$, $S(1,0) = [1,0,1,1,2,\ldots]$, and $S(2,0) = [2,0,2,2,\ldots]$.

Input

Input consists of several cases, each with a different natural number n between 1 and 10^6.

Output

For every n, print the number of pairs (a,b) such that n appears at a position $i \geq 3$ in $S(a,b)$.

Hint

Depending on your solution, Cassini's identity could be useful: $F_{i-1} \cdot F_{i+1} - F_i^2 = (-1)^i$.

Sample input

```
2
1
3
9
10
1000
1000000
```

Sample output

```
3
1
4
8
10
780
773883
```

Problem information

Author : Salvador Roura
Generation : 2024-04-30 17:52:16