Stealthy Number Test • Mark Senn • last updated 2021-12-18 21:57-05

Problem Statement

From The Weekly Challege - 143 Task #2: Stealthy Number retrieved on 2021-12-18 at 19:27-05:

Submitted by: Mohammad S Anwar

You are given a positive number, $n.

Write a script to find out if the given number is Stealthy Number .

> A positive integer N is stealthy, if there exist positive integers a, b, c, d such that a * b = c * d = N and a + b = c + d + 1.

Example 1
```
Input:  $n = 36
Output:  1
Since 36 = 4 (a) * 9 (b) = 6 (c) * 6 (d) and 4 (a) + 9 (b) = 6 (c) + 6 (d) + 1
```
Example 2
```
Input:  $n = 12
Output:  1
Since 2 * 6 = 3 * 4 and 2 + 6 = 3 + 4 + 1
```
Example 3
```
Input:  $n = 6
Output:  0
Since 2 * 3 = 1 * 6 but 2 + 3 != 1 + 6 + 1
```

Raku Solution

I like Raku much better than Perl. One reason: more expressive programming operators.

```
# Read $n.
my $n = $*IN.get;

# Get divisors of $n from 1 to sqrt($n).  If $n is 36 then @d1 is (1, 2, 3, 4, 6).
my @d1 = (1 .. sqrt $n).grep({$n %% $_});

# Get divisors of $n from $n down to sqrt($n).  If $n is 36 then @d2 is (36, 18, 12, 9, 6).
my @d2 = $n <</<< @d1;

# Add divisors in @d1 and @d2 together.  If $n is 36, @sum is (37, 20, 15, 13, 12).
my @sum = @d1 <<+>> @d2;

# If there any @sum entries that differ by one print "1", otherwise print "0".
# If $n is 36, the @sum entries 12 and 13 differ by one.
say (@sum.any == @sum.any+1) ?? 1 !! 0;
```

Wolfram Language Solution

Note how easy it is to translate the problem statement into the solution.

```
n = Input[];
solution = Solve[
    a > 0 && b > 0 && c > 0 && d > 0 && a b == c d && c d == n && a + b == c + d + 1,
    {a, b, c, d},
    Integers
];
If [solution == {}, Print[0], Print[1]];
```