code jam
print "hello, world!"

Qualification Round 2016

[A. Counting Sheep](#)

[B. Revenge of the Pancakes](#)

[C. Coin Jam](#)

[D. Fractiles](#)

Contest Analysis

[Questions asked](#) `1`

− Submissions

Counting Sheep

| 7pt | Not attempted **26558/29356 users** correct (90%) |
| 8pt | Not attempted **25729/26216 users** correct (98%) |

Revenge of the Pancakes

| 10pt | Not attempted **22527/23686 users** correct (95%) |
| 10pt | Not attempted **21383/22147 users** correct (97%) |

Coin Jam

| 10pt | Not attempted **13361/15342 users** correct (87%) |
| 20pt | Not attempted **6297/9111 users** correct (69%) |

Fractiles

| 10pt | Not attempted **8250/9708 users** correct (85%) |
| 25pt | Not attempted **2356/4955 users** correct (48%) |

− Top Scores

Lewin	100
Endagorion	100
xiaowuc1	100
xyz111	100
HellKitsune123	100
seanwentzel	100
ivan.popelyshev	100
burunduk3	100
Nicolas16	100
ctunoku	100

Practice Mode   [Contest scoreboard](#) | [Sign in](#)

Contest Analysis

[Overview](#) | Problem A | [Problem B](#) | [Problem C](#) | [Problem D](#)

Counting Sheep: Analysis

The most natural approach to this problem is to simulate the process: keep track of which digits Bleatrix has seen so far, and then keep generating and checking the numbers she names until she has seen all of the digits 0-9. But are there other cases like **N** = 0 that go on forever or take an unacceptably long time?

One way to answer this question for the purposes of the problem to just check all possible cases between 0 and 10^6 before downloading the Large dataset. With a well-written program on a reasonably fast machine, this should take only a few seconds.

More generally, it can be proven that for any **N** > 0, the sheep doesn't have to keep naming numbers for very long:

- Regarding the digit 0: The tenth number that Bleatrix names is 10 times **N**, and is therefore guaranteed to end in 0.
- Regarding digits 1-9: Consider the smallest power of 10 greater than **N**; call it P. Once the process reaches a number at least as large as P, then the leftmost digit will take on every possible value from 1-9 as the number increases up to (or past) $9P$. No digit can be skipped, because that would require the step between successive numbers (which equals **N**) to be larger than P... but we know that **N** is less than P because of how we chose P.

Since, by definition of P, 10**N** ≥ P, we will reach a number larger than P after naming at most 10 numbers, and we will reach a number greater than $9P$ after naming at most 90 numbers. So, after checking for the special case of **N** = 0, we can run our simulation without fear of running forever, running for long enough to run out of time, or overflowing even a 32-bit integer. Within the limits of the Small and Large datasets, the worst cases turn out to be 125 followed by any number of zeroes. In those cases, Bleatrix will name 72 numbers before falling asleep.

All problem statements, input data and contest analyses are licensed under the Creative Commons Attribution License.

© 2008-2017 Google   Google Home - Terms and Conditions - Privacy Policies and Principles

Powered by

Google Cloud Platform