FiveThirtyEight's March 11, 2022 Riddler

Emma Knight

March 21, 2022

This week's riddler, courtesy of Rolfe Petschek, is about perplexing postal workers[1]:

Question 1. *A postal worker and his customer joke about the various ways the customer could mathematically encode her post office box number.*

The customer realizes that every integer greater than 1 can be encoded via at least one Fibonacci-like sequence using an ordered triple (m, n, q). The encoded number is the qth member of the sequence after the first two positive integers m and n, where each term is the sum of the previous two terms. For example, 7 has the encodings $(3, 4, 1)$ and $(1, 3, 2)$.

In an attempt to stump the postal worker, the customer prefers encodings with a maximal value of q. What encoding should she use for the number 81?

Extra Credit*: What encoding should she use for the number 179?*

While it is entirely possible to just write code and find the answer, there is a nicer way. Instead of starting a sequence at god knows where and trying to land at some integer a, we will start at a and see how far back we can make the sequence go before it's no longer positive. Thus, we will consider a sequence $a_0, a_1, \ldots$ where $a_0 = 1$ and $a_i = a_{i+1} + a_{i+2}$, that is, $a_{i+2} = a_i - a_{i+1}$. Writing $a_1 = k$, one immediately sees what this sequence is: $a, k, a-k, 2k-a, 2a-3k, 5k-3a, \ldots$. The coefficients in front of a and k in the i^{th} term are $(-1)^i F_i$ and $(-1)^{i+1} F_{i+1}$ ($F_0 = 1, F_1 = 0$, which is not close to the standard indexing but I don't care).

In order for a_i to be positive, one needs $(-1)^i F_i a + (-1)^{i+1} F_{i+1} k > 0$. If i is even, this becomes $\frac{F_i}{F_{i+1}} a > k$, and if i is odd, this becomes $\frac{F_i}{F_{i+1}} a < k$. Thinking about what it means for requiring all of the terms to be positive, this gives a sequence of successively shrinking intervals that you can compute by hand, and one needs an integer in all of them. You run into issues when there are no integers in these intervals.

While that is enough to start computing things, pure thought can tell you what the best guesses are for k: since these intervals will close in on $\frac{a}{\phi}$, you just try k as one of the two integers closest to $\frac{a}{\phi}$ and see how far back you can go. $\frac{81}{\phi} \approx 50.0608$, so it's pretty clear that the best choice for k

[1] Please don't do this to them, they have enough work as is.

is 50. The sequence continues $81, 50, 31, 19, 12, 7, 5, 2, 3, -1, \ldots$, so one gets that the best encoding is $(3, 2, 7)$.

For the extra credit, $\frac{179}{\phi} \approx 110.6281$, so it's not immediately clear whether you want 111 or 110. Giving both sequences, one gets $179, 111, 68, 43, 25, 18, 7, 11, -4$ and $179, 110, 69, 41, 28, 13, 15, -2$. Thus, the best encoding is $(11, 7, 6)$.

For completeness, here is the code I wrote in python to see the answer:

```
##target is the number that is trying to be encoded.
##best is the best value of q.  winning is the best
##triple (a, b, q)

target = 81
best = 0
winning = []

##This code just loops over all possibilities of a and b
##and looks for ones that hit q.  If it finds a hit, it
##checks to see if this is a better hit and if so updates
##best and winning.

for a in range(1, target):
    for b in range(1, target):
        x = a
        y = b
        q = 0
        while(y < target):
            z = x
            x = y
            y = z + y
            q += 1
        if y == target:
            if q > best:
                best = q
                winning = [a, b, q]

print(target, winning)
```