Advent of code day 6 closed solution

Preface:

This proof follows the sames ideas of the proof of the closed solution to the Fibonacci function which this problem recalls.

Definitions:

We are looking for the closed form of $f(n)$ which is defined as the number of fish with 0 days left before creating another fish after n steps if we start with only one fish with 6 days left in the beginning.

There are two ways for a fish to end up with 0 days left after n steps:

1. Spawn 8 days before with 8 days left: there are $f(n-9)$ fish like this because this is the number of fish ready to spawn new fish 9 days before.
2. Spawn another fish 6 days before and reset the timer to 6: there are $f(n-7)$ fish like this because this is the number of fish ready to spawn new fish 7 days before.

With this information and remembering that there is one fish starting with 6 days left we can define $f(n)$ mathematically

$f(0)=0,\ f(1)=0,\ f(2)=0,\ f(3)=0,\ f(4)=0, f(5)=0,\ f(6)=1,\ f(7)=0,\ f(8)=0$

$f(n)=f(n-7)+f(n-9)$

Generating function:

$F(z):=\sum_{n=0}^{\infty} f(n)z^n$

Calculation:

$F(z)=f(0)+f(1)z+f(2)z^2+f(3)z^3+f(4)z^4+f(5)z^5+f(6)z^6+f(7)z^7+f(8)z^8+$
$+\sum_{n=9}^{\infty}[f(n-7)+f(n-9)]z^n$

$F(z)=z^6+\sum_{n=9}^{\infty}[f(n-7)]z^n+\sum_{n=9}^{\infty}[f(n-9)]z^n$

$F(z)=z^6+z^7\sum_{n=9}^{\infty}[f(n-7)]z^{(n-7)}+z^9\sum_{n=9}^{\infty}[f(n-9)]z^{(n-9)}$

$\sum_{n=9}^{\infty}[f(n-9)]z^{(n-9)}=\sum_{n=0}^{\infty} f(n)z^n=F(z)$

$\sum_{n=9}^{\infty}[f(n-7)]z^{(n-7)}=-f(7)-f(8)z+\sum_{n=7}^{\infty}[f(n-7)]z^{(n-7)}=F(z)$

$F(z)=z^6+z^7F(z)+z^9F(z)$

$F(z)=\frac{z^6}{1-z^7-z^9}$

Decomposing:

We use `partial_fractions.py` to decompose $F(z)$ as a sum of simpler fractions of the form $\frac{a}{z+b}$

$$F(z) = \frac{a_1}{z-b_1} + \frac{a_2}{z-b_2} + \frac{a_3}{z-b_3} + \frac{a_4}{z-b_4} + \frac{a_5}{z-b_5} + \frac{a_6}{z-b_6} + \frac{a_7}{z-b_7} + \frac{a_8}{z-b_8} + \frac{a_9}{z-b_9}$$

The program output are the vectors:

```
numerators = [a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9]
zeros = [b_1, b_2, b_3, b_4, b_5, b_6, b_7, b_8, b_9]
```

The program:

Using the identity $\frac{a}{z-b} = \sum_{n=0}^{\infty} -ab^{-n-1}z^n$

$$F(z) = \sum_{n=0}^{\infty} f(n)z^n = \sum_{n=0}^{\infty}\sum_{i=1}^{9} -a_i b_i^{-n-1} z^n$$

By equaling the coefficients of the same power of z we obtain:

$$f(n) = \sum_{i=1}^{9} -a_i b_i^{-n-1} z^n$$

which is what `func(n)` evaluates in `closed_solution.py`

Finding the fish:

Now that we know the number of fish with 0 days left how do we find the total number of fish?

Let's name the number of fish with a certain day left number on each step starting from $n-8$:

Steps done	8	7	6	5	4	3	2	1	0
n-8	a	b	c	d	e	f	g	h	i
n-7	i	a	b+i	c	d	e	f	g	h
n-6	h	i	a+h	b+i	c	d	e	f	g
n-5	g	h	i+g	a+h	b+i	c	d	e	f
n-4	f	g	h+f	i+g	a+h	b+i	c	d	e
n-3	e	f	g+e	h+f	i+g	a+h	b+i	c	d
n-2	d	e	f+d	g+e	h+f	i+g	a+h	b+i	c
n-1	c	d	e+c	f+d	g+e	h+f	i+g	a+h	b+i
n	b+i	c	d+b+i	e+c	f+d	g+e	h+f	i+g	a+h

The last row represent the number of fish after n steps:

$$fish(n) = a + 2b + 2c + 2d + 2e + 2f + 2g + 2h + 3i$$

$$fish(n) = (a+h) + 2(b+i) + 2c + 2d + 2e + 2f + 2g + h + i$$

$$fish(n) = f(n) + 2f(n-1) + 2f(n-2) + 2f(n-3) + 2f(n-4) + 2f(n-5) +$$
$$+2f(n-6) + f(n-7) + f(n-8)$$

With this function we can calculate the total number of fish

Solving the puzzle:

Our function assumes one fish with 6 day left in the beginning, what if this is not the case?

If we start with a fish with days left x between 0 and 6 we can pretend it is a fish starting with 6 days left but $6 - x$ days are already passed.
To calculate the number of fish after n steps we can shift the function like this: $fish(n + 6 - x)$

To solve the puzzle we can calculate the $fish$ shifted function for each number in the input and sum the results.