[CS 11 25.1] Lab 3f – Aquarium 2

Cheatsheet is available here: https://oj.dcs.upd.edu.ph/cs11cheatsheet/

Submit solution [CS 11 25.1] Lab Exercise 3

My submissions

✔ **Points:** 150 (partial)

Time limit: 4.0s

Memory limit: 2G

› **Problem type**

⌄ **Allowed languages**
py3

Problem Statement

You decide to add more layers to your brand-new aquarium!

Given your brand-new aquarium, how many fish are in each layer?

Note. See Problem 3e for context.

Task Details

Your task is to implement a function named `num_fish`, which should have the following *signature*:

```
def num_fish(aquarium):
```

Copy

The above says that it has one argument `aquarium`. This is a string (`str`) that consists of several lines. If we number the lines as $1, 2, 3, \cdots$ starting from the top:

- The odd-numbered lines are of the form `+--- ... ---+` and separate two consecutive layers of the aquarium; that is, they start and end with `+` characters, and these `+` characters are separated by one or more `-` characters.
- The even-numbered lines start and end with `|` characters and are the actual layers of the aquarium.

The function must return a `list` of integers (`int`s), where the k^{th} integer (1-indexed) should correspond to the number of fish in the k^{th} layer from the top.

Restrictions

- The following symbols can now be used: `list`, `set`, `dict`, `enumerate`, `append`, `pop`, `extend`, `remove`, `sort`, `sorted`, `insert`, `clear`, `reverse`, `reversed`.
- Loops are allowed.
- Recursion is *disallowed*.
- Comprehensions are *disallowed*.
- Your source code must have at most $1{,}600$ bytes.

Examples

Example 1 Function Call

```
num_fish("""\
+-------------------+
|   <^^)))><        |
+-------------------+
""")
```

Copy

Example 1 Return Value

```
[1]
```

Copy

Example 2 Function Call

```
num_fish("""\
+--------------------------+
|    <^^)))><              |
+--------------------------+
|                          |
+--------------------------+
| ><(((^^>  <^^)))><  _()_ |
+--------------------------+
""")
```

Copy

Example 2 Return Value

```
[1, 0, 2]
```

Copy

Constraints

- The function `num_fish` will be called at most 100 times.
- `aquarium` is a string of printable ASCII characters with at most 50 lines.
- Each line of `aquarium` consists of at most 50 characters.
- `aquarium` follows the format given in **Task Details**.

Scoring

Note: New tests may be added and all submissions may be rejudged at a later time. (All future tests will satisfy the constraints.)

- You get 20 ❤ points if you solve all test cases where:
    - All fish are facing left.
    - The aquarium only contains fish.
- You get 15 ❤ points if you solve all test cases where:
    - All fish are facing left.
- You get 15 ❤ points if you solve all test cases where:
    - The aquarium only contains fish.
- You get 100 ● points if you solve all test cases.

Clarifications

Report an issue

No clarifications have been made at this time.