[CS 11 25.1] HOPE 3 – Messages

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

Submit solution [CS 11 25.1] HOPE 3

My submissions

Points: 190 (partial)
Time limit: 12.0s
Memory limit: 2G

Problem type

Allowed languages
py3

Problem Statement

Upon waking up, you are greeted with a bunch of messages from your friends.

You want to read some of the most recent messages already, but you also don't want to leave the older messages hanging. To fix this, you want to solve the following problem:

> Given a time t, what is the earliest message that came at time t or later?

If there are ties, the message with the shortest length is prioritized. If there are still ties, the message whose sender's name comes first alphabeticallly is prioritized.

Task Details

You may import the dataclass `Message` from `oj`. The implementation of `Message` is as follows:

```
from dataclasses import dataclass

@dataclass(frozen=True)
class Message:
    timestamp: int
    sender: str
    contents: str
```

Your task is to implement a function named `phone`, which should start like this:

```
def phone(messages):
```

Here, `messages` is a sequence of `Message` s.

The function must return a **function** which takes in an integer t and returns the correct `Message` as described in the problem statement. Return the `Message` object itself, not a copy. If there is no such message, return `None` instead.

Restrictions

Your source code must have at most $1{,}000$ bytes.

Example Testing

```
f = phone([
    Message(timestamp=20, sender="Nozomi", contents="Sensei"),
    Message(timestamp=20, sender="Fuuka", contents="Sensei."),
    Message(timestamp=30, sender="Sakurako",
contents="Sensei."),
    Message(timestamp=30, sender="Shiroko", contents="Sensei."),
    Message(timestamp=40, sender="Hikari", contents="hikari"),
])

assert f(15).sender == "Nozomi"
assert f(25).sender == "Sakurako"
assert f(35).sender == "Hikari"
```

Constraints

Let t be the number of times `phone` is called, let n be the number of times the functions returned by `phone` are called, and let m be the total number of messages across all calls to `phone`.

- $t \le 70{,}000$
- $n \le 350{,}000$
- $m \le 350{,}000$
- Each message's sender and contents is a string of ASCII characters with length at most 8.
- Each timestamp is an integer between 0 and 10^{12}, inclusive.
- No two messages with equal timestamps and message lengths have the same sender name.

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 100 ❤ points if you solve all test cases where:
  - $t \le 200$
  - $n \le 800$
  - $m \le 800$
- You get 35 ❤ points if you solve all test cases where:
  - $t \le 3{,}000$
  - $n \le 12{,}000$
  - $m \le 12{,}000$
- You get 55 ● points if you solve all test cases.

Clarifications

Report an issue

No clarifications have been made at this time.