[CS 11] Prac 9c – Logging

Problem Statement

At the end of every month, the security guard at UP Diliman's Quiwa Hall needs to produce a report detailing all the distinct individuals that entered the building at some point during that month.

To honor Sir Quiwa himself, you have decided to help the security guard by automating this process.

Given the sequence of lines in the log book, each containing a name and a "purpose", please collect all the distinct names, and for each one, provide a list of all their entries in the log book, as well as the corresponding line numbers. Please provide the lists in order of increasing line number.

In this problem, line numbering starts from 1.

Task Details

Your task is to implement a function called `entry_report`. This function has a single parameter, a sequence of pairs of `str`s, each representing a line in the log book. In each pair:

- The first element is the name; and
- The second element is the purpose.

The sequence can either be a `tuple` or a `list`.

The function must return a `dict` whose keys are the names (`str`s) and each corresponding value is a `list` of `int`s denoting the line numbers where that name appears.

In this problem, line numbering starts from 1.

Restrictions

(See 9a for more restrictions)

For this problem in particular:

- Recursion is **disallowed**. (The recursion limit has been greatly reduced.)
- The source code limit is 2000.

Example Calls

Example 1 Function Call

```
entry_report([
    ('Kevin', 'Class'),
    ('Rich', 'Class'),
    ('Jem', 'Research'),
    ('Kevin', 'Aircon'),
    ('Kevin', 'Aircon'),
    ('Jem', 'Aircon'),
    ('Kevin', 'Class kunwari'),
    ('Jozelle', 'Class'),
    ('Kevin', 'Aircon'),
])
```

Copy

Example 1 Return Value

```
{
    'Jem': [3, 6],
    'Jozelle': [8],
    'Kevin': [1, 4, 5, 7, 9],
    'Rich': [2],
}
```

Copy

Constraints

- The function `entry_report` will be called at most $60{,}000$ times.
- The total length of all input sequences will be at most $200{,}000$.
- The input sequence will be at most $200{,}000$ elements.
- Each name is a nonempty string of at most 7 English letters.
- Each purpose is a string of at most 14 English letters or spaces.

Scoring

- You get 40 ♥ points if you solve all test cases where:
  - the total length of all input sequences is at most 500.
  - the input sequence has at most 50 elements.
- You get 35 ♥ points if you solve all test cases where:
  - the total length of all input sequences is at most $8{,}000$.
  - the input sequence has at most $4{,}000$ elements.
- You get 100 ♥ points if you solve all test cases.

Clarifications

Report an issue

No clarifications have been made at this time.

Submit solution

[CS 11] Practice 9 ♥

My submissions

✓ **Points:** 175 (partial)

Time limit: 4.0s

Memory limit: 1G

Author: kvatienza (Kevin Atienza)

Problem type

Allowed languages
~~NONE~~, py3