

[CS 11] Prac 3g – Attendance

oj.dcs.upd.edu.ph/problem/cs11prac3g

[Submit solution](#)

[My submissions](#)

Points: 125 (partial)

Time limit: 4.0s

Memory limit: 1G

Author:

[kvatienda \(Kevin Atienza\)](#)

Problem type

Allowed languages

NONE, py3

Problem Statement

One of the things I learned is that encoding attendance sheets and figuring out who is present or absent every meeting is very annoying. Let's automate it!

Given the class list and the attendance list, determine who were present and who were absent.

Task Details

Your task is to implement a function called `attendance`. This function has two parameters:

- `attended` — a `tuple` of `strs` denoting the people who attended today's class.
- `class_list` — a `tuple` of `strs` denoting the people who are enrolled in the class.

The function must return a *pair* of `tuples`, each consisting of `strs`:

- The first `tuple` must contain the people who attended;
- The second `tuple` must contain the people who didn't attend.

Please return the elements in the order they appear in the class list.

Restrictions

For this problem:

- Recursion is allowed.
- Up to 88 functions are allowed.
- Comprehensions are **disallowed**.
- The `range`, `min`, `max`, and `sum` symbols are **disallowed**.
- The source code limit is 800800.

Example Calls

Example 1 Function Call

Copy

```
attendance(  
    ('dennis', 'jeremiah', 'eugene'),  
    ('eugene', 'alfred', 'dennis', 'vincent'),  
)
```

Example 1 Return Value

Copy

```
(  
    ('eugene', 'dennis'),  
    ('alfred', 'vincent'),  
)
```

Example 1 Explanation

Note that `jeremiah` attended the class but isn't in the class list.

Constraints

- The function `attendance` will be called at most 200200 times.
- The attendance list consists of at most 2525 distinct names.
- The class list consists of at most 5050 distinct names.
- Each name is nonempty and consists of up to 88 lowercase English letters.

Scoring

- You get 125125 ❤ points if you solve all test cases.

[Report an issue](#)

Clarifications

No clarifications have been made at this time.