



[CS 11 25.1] HOPE 1 A4 – Strategic Planning 2

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

Problem Statement

(See Problem A3 for context.)

Now that you're familiar with synergies, let's apply that to a boss battle!

You have a list of students you're planning to use for a battle. Each student has an attack type and an armor type, while the boss also has its own attack type and armor type.

You want to figure out three things:

- Which students would make for good *attackers* for this boss battle? For a student to be a good attacker, their attack type's multiplier against the boss' armor type should be `Weak` or `Effective`.
- Which students would make for good *defenders* for this boss battle? For a student to be a good defender, the boss' attack type's multiplier against the student's armor type should be `Resist`.
- Which students would make for both good *attackers* and good *defenders*?

Submit solution
[CS 11 25.1]
HOPE 1
My submissions

✓ Points: 125 (partial)

⌚ Time limit: 12.0s

≡ Memory limit: 2G

➤ Problem type

▼ Allowed languages

py3

Task Details

Your task is to implement a function named `students_to_use`. This function has two parameters:

- The first parameter is a tuple of triples. Each triple consists of three strings, denoting a student's name, attack type, and armor type, respectively.
- The second parameter is a pair of strings indicating the boss' attack type and armor type.

The function must return three tuples of strings:

- The first tuple must contain the names of all students who would make for good attackers.
- The second tuple must contain the names of all students who would make for good defenders.
- The third tuple must contain the names of all students who would make for both good attackers and good defenders.

In each tuple, the names must be ordered in the same way as they appear in the input.

Restrictions

- Your source code must have at most 2200 bytes.

Examples

Example 1 Function Call

```
students_to_use((  
    ("Nozomi", "Sonic", "Light"),  
    ("Mika", "Piercing", "Light"),  
    ("SMika", "Sonic", "Special"),  
    ("Hikari", "Sonic", "Light"),  
, ("Piercing", "Elastic"))
```

Copy

Example 1 Return Value

```
(  
    ("Nozomi", "SMika", "Hikari"),  
    ("Nozomi", "Mika", "Hikari"),  
    ("Nozomi", "Hikari"),  
)
```

Copy

Constraints

- The function `students_to_use` will be called at most 200 times.
- Each student's name consists of at most 11 uppercase or lowercase English letters.
- Each attack type is one of `Normal`, `Explosive`, `Piercing`, `Mystic`, and `Sonic`.
- Each armor type is one of `Normal`, `Light`, `Heavy`, `Special`, and `Elastic`.
- No student appears more than once.
- You have at most 200 students.

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 25 ❤ points if you solve all test cases where:
 - No attack type is `Sonic`.
 - No armor type is `Elastic`.

- You get 100 🎯 points if you solve all test cases.

Clarifications

Report an issue

No clarifications have been made at this time.