

[CS 11 25.1] Lab 2f – Study Buddies 2

Problem Statement

Out of the n members of the Supplementary Lessons Club, $n - 1$ of them have just received their results in their m quizzes. Each quiz is scored an integer between 0 and 100 inclusive.

Seeing these results, Hanako is now deciding what scores she should get. For each quiz, she wants to get the minimum score needed to make the average score of the n members in that quiz at least 60.

What score should Hanako get in each quiz?

If this is not possible for some quizzes, say so (see **Task Details**).

Task Details

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

```
def scores_needed(results):
```

Copy

The above says that it has one argument `results`, which is a `tuple` of $n - 1$ tuples denoting the quiz scores of the $n - 1$ members other than Hanako. Each tuple consists of m integers (`int`s) denoting the scores of a member in the m quizzes.

The function must return a `tuple` of m integers (`int`s) denoting the minimum score Hanako needs in each quiz to make the average score of all n members ≥ 60 for that quiz.

If this is not possible for a certain quiz, return `None` for its entry instead of an integer.

Restrictions

- The following symbols can be used: `min`, `max`, `sum`, `range`, `all`, `any`, `sorted`.
- recursion is allowed.
- comprehensions are allowed.
- at most 12 functions can be defined.
- Your source code must have at most 1,200 bytes.

Examples

Example 1 Function Call

```
scores_needed((
    (40, 80, 10),
    (90, 10, 0),
))
```

Copy

Example 1 Return Value

```
(50, 90, None)
```






Copy

Constraints

- The function `scores_needed` will be called at most 500 times.
- $2 \leq n \leq 30$
- $1 \leq m \leq 30$
- Each score is between 0 and 100 inclusive.

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 15  points if you solve all test cases where:
 - $n = 2$
 - $m = 1$
- You get 15  points if you solve all test cases where:
 - $n = 2$
- You get 15  points if you solve all test cases where:
 - $m = 1$
- You get 15  points if you solve all test cases where:
 - $n \leq 6$
 - $m \leq 6$
- You get 50  points if you solve all test cases.


? Clarifications


No clarifications have been made at this time.


Report an issue


Submit solution

 **Points:** 110 (partial)

 **Time limit:** 4.0s

 **Memory limit:** 2G

 **Problem type**

 **Allowed languages**
py3