

[CS 11 25.1] Mock HOPE 2g – Beat the Heat

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

Problem Statement

In an alternate universe...

It has been getting really hot these days! At this rate, you can cook an egg outside...

You know what the heat index in your area has been over the past n days (including today). In order to guess what the heat index will be tomorrow, you decide to use a *linear regression model*.

A *simple linear regression* model basically takes in a bunch of points and, as the name suggests, tries to fit a **line** to those points. Typically, the line is chosen so that the sum of the squares of the errors (how far the line's approximation is from the actual data point) is as small as possible.

Formally, if the line is $f(x) = mx + b$ and the n data points are $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$, then you want to minimize the value of

$$\sum_{i=1}^n (y_i - f(x_i))^2.$$

Once you obtain the line, you can then use it to *predict* the corresponding y -values for x -values not necessarily in your dataset.

For our purposes, we give an x -value of a to a date a days in the future. Note that a may be negative.

Given the heat indices over the past n days (including today), can you guess the heat index tomorrow?

By the way, you may use Python's builtin `statistics` module, which has a function that can compute the simple linear regression parameters.

Task Details

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

```
def guess_heat_index(previous):
```

Copy

The above says that it has one argument `previous`. This is a list of n integers (`int`s) denoting the heat indices over the past n days.

The function must return a `float` denoting the predicted heat index tomorrow.

Important: Answers that are "close" enough will also be accepted. Formally, if the answer your implementation of `guess_heat_index` provides is x and the judge's answer is x' , then it will be considered correct if $|x - x'| < 10^{-8}$.

Restrictions

- The following symbols can be used:
 - `list`, `set`, `dict`, `enumerate`, `append`, `pop`, `extend`, `remove`, `sort`, `sorted`, `insert`, `clear`, `reverse`, `reversed`, `iter`, `next`, `zip`.
- The following imports are allowed:
 - `count` and `islice` from `itertools`.
 - most functions from `statistics`.
- Loops are allowed.
- Generators and comprehensions are allowed.
- Recursion is allowed.
- Your source code must have at most 400 bytes.

Examples

Example 1 Function Call

```
guess_heat_index([36, 38, 40])
```

Copy

Example 1 Return Value

```
42.0
```

Copy

Example 2 Function Call

```
guess_heat_index([50, 48, 45])
```

Copy

Example 2 Return Value

```
42.666666666666666666
```

Copy

Example 3 Function Call

```
guess_heat_index([50, 48, 45, 60, 40, 20, 90, 55, 60, 64])
```

Copy

Example 3 Return Value

```
64.2
```


Copy

Constraints

- The function `guess_heat_index` will be called at most 200 times.
- $2 \leq n \leq 100$
- Each given heat index is an integer between 0 and 60 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 200  points if you solve all test cases.

? Clarifications

No clarifications have been made at this time.

Report an issue

Submit solution

[CS 11 25.1]

Mock HOPE 2g

[My submissions](#)

- ✔ **Points:** 200 (partial)
- ⌚ **Time limit:** 6.0s
- 📦 **Memory limit:** 2G

- **Problem type**
- ▼ **Allowed languages**
py3