

[CS 11 25.1] HOPE 2b – Vibe Check

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

Problem Statement

You plan to watch a movie for the upcoming break, but you're not sure about what you should watch! You want to watch something that *passes the vibe check*: something that you can talk about with other people. And what better way to check if you've got the right movie than to look at its ratings?

In your area, people rate movies from 0 to 10^{20} . We say that a movie *passes the vibe check* if there are **exactly two** ratings that appear most often among all ratings for that movie.

Given the ratings for a certain movie, does it pass the vibe check?

Task Details

Your task is to implement a function named `passes_vibe_check`, which should start like this:

```
def passes_vibe_check(ratings):
```

Copy

Here, `ratings` is a list of integers, denoting the ratings for the movie.

The function must return a Boolean denoting whether or not the movie passes the vibe check.

Restrictions

- Loops and lists are allowed.
- Sets and dictionaries are allowed.
- Recursion is **disallowed**. (The recursion limit has been greatly reduced.)
- Generators and comprehensions are allowed.
- Your source code must have at most 450 bytes.

Examples

Example 1 Function Call

```
passes_vibe_check([10, 0, 0, 10, 1])
```

Copy

Example 1 Return Value

```
True
```

Copy

Example 1 Explanation

There are exactly two values that appear the most frequently (10 and 0), so this movie passes the vibe check.

Example 2 Function Call

```
passes_vibe_check([1000, 9, 7, 8, 1000, 8, 7])
```

Copy

Example 2 Return Value

```
False
```

Copy

Example 2 Explanation

There are exactly three values that appear the most frequently (1000, 7, and 8), so this movie does not pass the vibe check.



Constraints

Let n be the number of ratings given in a single call to `passes_vibe_check`.

- The function `passes_vibe_check` will be called at most 70,000 times.
- $1 \leq n \leq 350,000$
- The sum of the n s across all test cases is at most 350,000.
- Each rating is between 0 and 10^{20} , 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 90  points if you solve all test cases where:
 - $1 \leq n \leq 50$
 - The sum of the n s across all test cases is at most 100.
 - Each rating is between 0 and 100, inclusive.
- You get 140  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]



HOPE 2

My submissions

 **Points:** 230 (partial)

 **Time limit:** 9.0s

 **Memory limit:** 2G

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