

Problem 1086: High Five

Problem Information

Difficulty: Easy

Acceptance Rate: 74.23%

Paid Only: Yes

Tags: Array, Hash Table, Sorting, Heap (Priority Queue)

Problem Description

Given a list of the scores of different students, `items`, where `items[i] = [IDi, scorei]` represents one score from a student with `IDi`, calculate each student's **top five average**.

Return the answer as an array of pairs `result`, where `result[j] = [IDj, topFiveAveragej]` represents the student with `IDj` and their **top five average**. Sort `result` by `IDj` in **increasing order**.

A student's **top five average** is calculated by taking the sum of their top five scores and dividing it by `5` using **integer division**.

Example 1:

Input: `items = [[1,91],[1,92],[2,93],[2,97],[1,60],[2,77],[1,65],[1,87],[1,100],[2,100],[2,76]]`
Output: `[[1,87],[2,88]]` **Explanation:** The student with ID = 1 got scores 91, 92, 60, 65, 87, and 100. Their top five average is $(100 + 92 + 91 + 87 + 65) / 5 = 87$. The student with ID = 2 got scores 93, 97, 77, 100, and 76. Their top five average is $(100 + 97 + 93 + 77 + 76) / 5 = 88.6$, but with integer division their average converts to 88.

Example 2:

Input: `items = [[1,100],[7,100],[1,100],[7,100],[1,100],[7,100],[1,100],[7,100],[1,100],[7,100]]`
Output: `[[1,100],[7,100]]`

Constraints:

* `1 <= items.length <= 1000` * `items[i].length == 2` * `1 <= IDi <= 1000` * `0 <= scorei <= 100` * For each `IDi`, there will be **at least** five scores.

Code Snippets

C++:

```
class Solution {
public:
    vector<vector<int>> highFive(vector<vector<int>>& items) {

    }
};
```

Java:

```
class Solution {
    public int[][] highFive(int[][] items) {

    }
}
```

Python3:

```
class Solution:
    def highFive(self, items: List[List[int]]) -> List[List[int]]:
```