

PromptScratchpadSolutionsVideo Explanation

Difficulty: Category: Greedy AlgorithmsSuccessful Submissions: 30,603+

Class Photos

☆

It's photo day at the local school, and you're the photographer assigned to take class photos. The class that you'll be photographing has an even number of students, and all these students are wearing red or blue shirts. In fact, exactly half of the class is wearing red shirts, and the other half is wearing blue shirts. You're responsible for arranging the students in two rows before taking the photo. Each row should contain the same number of the students and should adhere to the following guidelines:

- All students wearing red shirts must be in the same row.
- All students wearing blue shirts must be in the same row.
- Each student in the back row must be strictly taller than the student directly in front of them in the front row.

You're given two input arrays: one containing the heights of all the students with red shirts and another one containing the heights of all the students with blue shirts. These arrays will always have the same length, and each height will be a positive integer. Write a function that returns whether or not a class photo that follows the stated guidelines can be taken.

Note: you can assume that each class has at least 2 students.

Sample Input

```
redShirtHeights = [5, 8, 1, 3, 4]
blueShirtHeights = [6, 9, 2, 4, 5]
```

Sample Output

```
true // Place all students with blue shirts in the back row.
```

Hints

Hint 1

Start by determining which row will have the students wearing blue shirts and which row will have the students wearing red shirts. Once you know this, how can you determine if it's possible to take the photo?

Hint 2

The shirt color of the tallest student will determine which students need to be placed in the back row. The tallest student can't be placed in the front row because there's no student taller than them who can be placed behind them.

Hint 3

Once you know which students should be placed in each row, you can simply check if each student in the back row can be paired with a student in the front row who is shorter than them. If you can't find a satisfactory pairing for every student in the back row, then you can't take the photo.

Hint 4

Sort each input array in descending order, then determine which students will be in the front and back rows following Hint #2. After this, simply loop through your sorted input arrays, and check if the current tallest student in the back row is taller than the current tallest student in the front row. If you find that the current tallest student (one that has yet to be placed) in the back row isn't taller than the current tallest student in the front row, then the photo can't be taken.

Your Solutions

Solution 1Solution 2Solution 3

```
1 def class_photo_possible(red, blue):
2     red.sort()
3     blue.sort()
4
5     if sum(red) > sum(blue):
6         for idx in range(len(red)):
7             if red[idx] < blue[idx] or red[idx] == blue[idx]:
8                 return False
9
10    else:
11        for idx in range(len(blue)):
12            if blue[idx] < red[idx] or blue[idx] == red[idx]:
13                return False
14
15    return True
16
17 def classPhotos(redShirtHeights, blueShirtHeights):
18     red = redShirtHeights
19     blue = blueShirtHeights
20     return class_photo_possible(red, blue)
21
22
23 # Kunal Wadhwa
24
25 # https://github.com/kunal5042
26 # https://leetcode.com/kunal5042/
27 # https://www.hackerrank.com/kunalwadhwa_cs
28 # https://www.linkedin.com/in/kunal5042/
29
```

TestsQuick TestSandbox

Test Case 1

```
1 {
2   "redShirtHeights": [5, 8, 1, 3, 4],
3   "blueShirtHeights": [6, 9, 2, 4, 5]
4 }
```

Test Case 2

```
1 {
2   "redShirtHeights": [6, 9, 2, 4, 5],
3   "blueShirtHeights": [5, 8, 1, 3, 4]
4 }
```

Test Case 3

```
1 {
2   "redShirtHeights": [6, 9, 2, 4, 5, 1],
3   "blueShirtHeights": [5, 8, 1, 3, 4, 9]
4 }
```

Test Case 4

```
1 {
2   "redShirtHeights": [6],
3   "blueShirtHeights": [6]
4 }
```

Test Case 5

```
1 {
2   "redShirtHeights": [126],
3   "blueShirtHeights": [125]
```

Custom OutputRaw Output

Submit Code

Yay, your code passed all the test cases!

15 / 15 test cases passed.

✓ Test Case 1 passed!

✓ Test Case 2 passed!

✓ Test Case 3 passed!

✓ Test Case 4 passed!

✓ Test Case 5 passed!

✓ Test Case 6 passed!

✓ Test Case 7 passed!

✓ Test Case 8 passed!