CS310 Python for Data Science

Assignment 1

In this assignment, you will design a function to help organize multiple meetings during a day. Each meeting will have a start time and end time. Each meeting will need to reserve a private conference room during that time. Your function will take all the meeting schedules and calculate the minimum number of conference rooms you will need. Here are the requirements:

1. The function will take the input as a sequence of lists, each list only contains two numbers, the start time and the end time. For simplicity, we assume the time is just a float number in the range of 0.0 to 24.0. For example, the function may be given the following input:

   findMinRooms([1.2, 3.4], [2.3, 5.0], [3.1, 8.0])

   Your code should return 3 as the result, since during this time window [3.1, 3.4], all three meetings will be going on in parallel. Another example:

   findMinRooms([1.2, 3.4], [2.3, 5.0], [4.1, 8.0])

   this will return 2. Another example:

   findMinRooms([1.2, 3.4], [2.3, 5.0], [3.1, 8.0], [1.0, 10.0])

   this will return 4.
2. Your function should be able to handle any number of arguments.
3. The sequence of meetings can be input in any order.
4. Your function should handle all wrong inputs gracefully, by catching all errors and display meaningful error messages. There should be no error/crash from the system.
5. Don't do brutal force implementation, i.e., by checking each meeting against every other meeting to count the time overlaps. Hint: You will need to re-organize and sort the data in some smart way.
6. You should be able to implement using only what we learned in class so far.
7. You are encouraged to discuss with other students about the high-level ideas, but never share anything at code level. You will have to write your own code independently. **Plagiarism will be detected and reported.**

Deliverable:

You will submit a single python (*.py) file using the Canvas assignment page before the due time. Other file types will not be accepted.

How do I grade:

I will use 10 test cases to test your function. Each test case will be 10 points. I will post the test cases after the due date. So please test thoroughly before you submit.

If you use the brutal force implementation, you will lose 10 points.

Total 100 points.