

## Programming Pre-Screen

Conference room scheduling. Find the nearest open conference room for a team in which a team can hold its meeting. Given  $n$  team members with the floor on which they work and the time they want to meet, and a list of conference rooms identified by their floor and room number as a decimal number, maximum number of people it fits and pairs of times they are open - find the best place for the team to have their meeting. If there is more than one room available that fits the team at the chosen time then the best place is on the floor the closest to where the team works.

E.g.

rooms.txt

7.11,8,9:00,9:15,14:30,15:00

8.23,6,10:00,11:00,14:00,15:00

8.43,7,11:30,12:30,17:00,17:30

9.511,9,9:30,10:30,12:00,12:15,15:15,16:15

9.527,4,9:00,11:00,14:00,16:00

9.547,8,10:30,11:30,13:30,15:30,16:30,17:30

Input:

5,8,10:30,11:30 # 5 team members, located on the 8th floor, meeting time 10:30 - 11:30

Output:

9.547

Please explain: how you solved the problem and how it would behave based on the different parameters (number of team members, longer meeting times, many rooms with random booking times). How would you test the program to ensure it always produced the correct results?

For extra credit, can you improve the solution to split the meeting across more than one room if say only one room is available for a fraction of the meeting and another room is free later to hold the remainder of the meeting during the set time. If you want to make this more powerful - assume that the number of room splits can happen in proportion to the length of the meeting so that say if a meeting is 8 hrs long then the algorithm could schedule it across say up to 4 rooms if a single room was not available for the whole time.

You may code the response in any programming language you like, however our primary DevOps programming languages are:

- Bash
- Perl
- Python
- Groovy