

FOR14

Algorithms and computer programming with Python

Autumn 2023

Practice assignment 3

The goal of this assignment is to help you start practicing for the final exam.

1 Problem description: car utilization follow up

After some tests with the prototype presented the car sharing company is interested now in develop a system the can scaled to manage the utilization of the cars. However, the following issues where found:

1. In some cases, the difference between the ending time of a reservation and the starting time for the next is such that there is no chance for that time to be used. In particular, if the difference is less than half an hour, that time spam is not practical for any user. Management is wondering how this should be accounted in the report.
2. The difference between starting time and ending time form some reservations is very short, as short as 1 minutes. The management also wants to consider that in the report.

The car sharing company wants you to use the same information of reservations made between March and August 2019. In particular, the manager wants to continue with the analysis of the cars with the ids 6, 7, 8, 9, 10.

2 Task

In order to make this scalable, they want now to have and OOP approach. The manager wants to be able to have a concept that provides an abstraction of the fleet. It must provide at least the functionality to compute the utilization of the whole fleet and to do the reshuffling. Also, they don't want to have the information of all the cars mixed. They want an abstraction to allow them to represent each car as an object with all its information and functionality encapsulated. Each car object has to allow the user at leas to compute the utilization of the car and a way to identified how much the original assignment and the assignment after reshuffling differ. Finally, they want the 2 new issues identified to be taken into account in the report.

All the code must be delivered in py files as part of this assignment and must be documented with proper comments. To obtain a complete you must submit:

- A clear explanation of your understanding and solution approach and a statement of your assumptions and scope of your solution. It is quite possible that your solution approach requires making some assumptions. Different solutions have varying degrees of efficiency. Additionally, you must provide the abstract design of each of your classes with an explicit specification of the data representation and the public interface.

- The pseudocode where needed, which must be implementable and must match your application.
- An implementation, ideally a running one. Keep in mind that for the final exam a running application will be part of the assessment.
- The code must be properly documented.
- The code must be an OOP implementation.

Have fun!