

Chapter 1

Algorithm Design

In this chapter the most interesting algorithm of myTaxiService are presented using pseudo-code (every developer can easily translate the pseudo-code into the desired programming language). In addition this pseudo-code is referred to an object-oriented programming language, like C++ or Java (the reader can think about the equivalent algorithm in non object-oriented programming language like C, so he has to create manually all the object data structures, the objects himself and he has to define a way to manage and save all the created objects).

The meaning of word *interesting*, used above to define the algorithms which we will present in this chapter, is the following: a characteristic and unique algorithm, used to implement a specific functionalities of this system. For instance, an algorithm to manage the backup of the database can be very complicated in describing the policies, the exceptions and the situations when execute its, but it is a common algorithm for all the systems that store data into a database.

In this chapter the algorithms analysed are the following:

- the city's areas creations and management;
- the queue's management;
- all the algorithms used to handle the special situation which occur when no cabmen are available in the area where the ride (which the Ride Allocator is no assigning) starting position is.

1.1 Map and Areas Creation Algorithms

TO BE FILLED

1.2 Queue Creator Algorithms

The Queue Creator is a subcomponent of the Ride Allocator¹. The queues' creation is an iterative process performed at the Ride Allocator creation and initialization. After the definition of the map and its areas, the Queue Creator is involved to create one queue into each area.

forall Area a *in* Map *do*
a.createQueue();

TO BE FILLED

1.3 Queue Manager Algorithms

TO BE FILLED

1.4 Ride Assignment Algorithm

TO BE FILLED

1.5 Special Algorithms

TO BE FILLED

¹see the ?? for a complete description.