Output:

**mechanic sarah available**

**name is ahmed**

**id is 1**

**age is 39**

**mechanic id is 0**

**appointment time is 7**

**mechanic sarah not available**

**mechanic john available**

**name is ahmed**

**id is 1**

**age is 39**

**mechanic id is 2**

**appointment time is 7**

**mechanic john available**

**name is dalal**

**id is 2**

**age is 20**

**mechanic id is 0**

**appointment time is 22**

**mechanic jack available**

**name is dalal**

**id is 2**

**age is 20**

**mechanic id is 2**

**appointment time is 22**

**mechanic sarah available**

**name is hamed**

**id is 3**

**age is 31**

**mechanic id is 1**

**appointment time is 22**

I created an abstract class called person with some generic data types like name, age, id. After that I created 2 inherited classes that are called mechanic and customer. There are specific attributes in each class like customer contains a mechanic id for the assigned mechanic while mechanic contains an array for his free time slots.

The next step was to define the functions and override the virtual function print info so the child classes would not turn out to be abstract classes. After finishing them I went on to the main.

In the main I created some mechanic objects and customer objects. The mechanic objects were then passed on to a queue and then inserted in a while loop with some random generated numbers to be compared with. The loops generated if the mechanics were free or not and if they were not free then the customers’ preferred time was then passed on to the next mechanic to check if he or she is free. If all mechanics are unavailable the application should generate a message saying that there are no available mechanics.

Errors: I did not use a template class queue as I was not able to create one, I am not sure how to check if the algorithm works or not, and I am not sure about the code’s logic.