Jolie Khaled Fawzy – 900211816

Assignment 2

```
Please enter the size of the array:
6
Customer Yassin should be assigned to the mechanic Mohanned at 1 : 30
Customer Nada should be assigned to the mechanic Khaled at 1 : 30
Customer Nala should be assigned to the mechanic Mohamed at 2 : 0
Customer Nour should be assigned to the mechanic Steven at 2 : 0
Customer Lujain should be assigned to the mechanic Karim at 3 : 30
Customer Rafik should be assigned to the mechanic Yosry at 12 : 0
```

As we can see, this screenshot is the output of our code that aims to create a car workshop appointment system, where the customers or the client can book appointment with the mechanic at as specific time. Of course, while working on this program, I faced many difficulties. For example, I was confused whether to use vectors or dynamic array to insert the size, but them decided to use dynamic array. Also, I couldn't know how to pass the value of the size, so I decided to create a dynamic array in the mechanic's constructer and add a variable called size. As we see in the output that the compiler asked me what is the size of the arrays, so I entered 6 as we have 6 customers. One of the other problems that I faced is that I could overload the operator <>=, it was hard to understand the concept of the work (this->). Also, creating a queue class was a little bit confusing as I started by #include <queue> which is much easier, but then discovered that I need to create a template class queue. Of course, some functions were non-void, but the function definition contains if else statements, so for the Boolean function for example, the return true and false were inside the braces of the if-else, so the compiler has considered it that it doesn't return any value although it is Boolean.

Let's now explain the output of the code. I already initialized all the value of the array of objects of type mechanic and customer. Also, I initialized the exact time that the customer wants to book the appointment at using the struct of appointments that contain as data members hours and minutes. So, I created objects of type appointment, and I started to initialize for every object the hour and the minutes. Then I create a for loop, that can check if the mechanic is available at that time specified by the customer. If yes, it will automatically assign him to the mechanic and if it's not, if the mechanic is already occupied with another customer at the same time, the customer will be assigned to another mechanic. Then I implemented the concept of bubble sort so that the time will be arranged. It swaps the time if for example customer 2 is before Customer 1. So, as we can see that on the output that the specific time is already ordered it starts by 1:30 am till 12:00 pm, as it's already 24 hours systems. The compiler has seen that the customer that

had specified the early time are Yassin and Nada, that's why they are the first customers in the output, then Nada and Nala and finally Rafik. In the code, I have specified that Yassin and nada has book an appointment at the same time at 1:30, which means that they can't be assigned to the same mechanic, that's why Yassin was assigned to mechanic Mohanned and Nada to mechanic Khaled. Same goes to Nala and Nour who booked at the same time, consequently having different mechanics. And finally the last customer who wanted to book at 12 pm, was assigned to the rest mechanic which is Yosry.