Programming Project 4

Jade Pearl

CMSC 215 6382 Intermediate Programming

Professor Romerl Elizes

**Approach**

As I moved through the project, I made sure to:

1. Reread chapter 13 to refresh my memory on the Comparable interface and to also reread chapter 19 to remember how to implement a generic class
2. Practiced creating the Interval class first since it was a generic class that implements the Comparable interface.
3. Defined the InvalidTime and Time classes
4. Worked through the Project4 class step by step. First, I made the text fields just as I had practiced in the previous project.
5. I implemented the buttons by calling the methods in the interval class but had to troubleshoot a lot of different bugs. I got the compare intervals to work, however I could not get the check time function to work.

**UML Diagram**

Below is the UML Diagram I created for this project:

A diagram of a project

Description automatically generated

**Test Plans**

Below are each of the 5 test cases I carried out to make sure that the project worked properly:

**Test 1: No input at all**

**A screenshot of a computer

Description automatically generated**

Result: when I clicked on the compare intervals or the check time buttons, I got a result of “Invalid time format” in my output box at the bottom because I had thrown an exception using that string as my message if nothing was input. This ensures that the user enters at least something.

**Test 2: Sample input with no Check Time**

A screenshot of a checker

Description automatically generated

Result: I got the expected output of the intervals overlapping with each other. This shows that my comparison of intervals works at least somewhat properly.

**Test 3: Sample input with Check time**

A screenshot of a checker

Description automatically generated

Result: I unfortunately did not get the anticipated result. My result was supposed to say that interval 2 contained the time of 12:50 PM. This shows that at least from the point of view of using a 12:00 as the check time does not work in my program.

**Test 4: Different input, no check time**

A screenshot of a checker

Description automatically generated

Result: I got the expected output that the intervals are disjoint, so this shows that my compare intervals functionality does work in more ways than one.

**Test 5: Different input, check time not using the 12 in the hours**

**A screenshot of a checker

Description automatically generated**

Result: Fortunately, when I use a time with an hour that is not 12, my check time functionality works. This shows that it does work in part, however I was unfortunately unable to figure out why it did not work when the hour for check time was 12. That seemed to be the only instance where it did not work.

**Lessons Learned**

This was a very long and difficult project. I devoted many an hour and effort into finding tutoring and into debugging my program. Although I could not get every aspect to work, I have learned a lot about how codes like this worked. By the end of this project, I learned:

1. How to implement generic classes with interfaces and how I can connect all these classes via the main method.
2. Better use of exception handling when it came with GUI and other objects that aren't in typical data type like the Time objects that I was using for exception handling.
3. Despite all of the hard work that I had done and the time I put into this project, sometimes we do not have the perfect finished result in the time we are given but we can still learn from it and try to understand what is going wrong
4. Better troubleshooting and debugging skills thanks to a classmate that taught me debugging in Visual Studio code for this project I was able to find breaking points in my code that the debugger could stop at while my program was running to see where something might have gone wrong despite no errors popping up. This includes uncaught exception handling.
5. Now learned how to better implement button handlers and connecting them to other methods outside of its class