Your Name: ______________________ **Student Number:** ______________

Lab 7: to be done in class individually or with a partner – due at the end of the lab

Part A - Debugging

Download the following files and load into a project in BlueJ:

- ClockDisplayDebug.java
- NumberDisplayDebug.java
- HistoricalMomentDebug.java
- MainDebug.java

The above classes have several bugs. When the main method in MainDebug is run, currently a null pointer exception is thrown.

Use System.out.println and/or the BlueJ debugger to troubleshoot and fix the problems in these classes.

Part B – Object References

Update DebugMain as follows:

```
ClockDisplay clockDisplayCommon = new ClockDisplay(10, 0);
ClockDisplay clockDisplayRef = clockDisplayCommon;
HistoricalMoment momentTest1 = new HistoricalMoment("Test Event 1", clockDisplayCommon);
HistoricalMoment momentTest2 = new HistoricalMoment("Test Event 2", clockDisplayRef);
```

1) Add additional code to prove that both historical moments are referencing the same clock display object. This can be done by printing the details of each historical moment object, adding a minute of time to one historical moment object (addMinuteToTimeOfEvent method) and then printing the details of each historical moment object again.
2) Print out your ClockDisplay and HistoricalMoment objects using System.out.println. What is output to the console? Which are the same and which are different?

Part C – Object Equality and this

Create a class called Person that has instance variables for the following data members:

- First name
- Last name
- Birth year
- Sex
- Weight in pounds
- Highest education level (can be "high school", "undergraduate", or "graduate")

Create one constructor takes all six values (above) as parameters and assigns them to the instance variables. No validation is required except for that described in the notes below.

Create setter methods for the First Name and Last Name instance variables.

NOTE: use "this" in the constructor and setter methods.

NOTE: do not store the first name "Mike" or last name "Mulder". If the constructor or setters gets those values as parameters, ignore it and set the name to null instead. Use the String methods .equals() or .equalsIgnoreCase() to compare Strings; do not use == to compare Strings (nor any objects).

Test your class to ensure that the constructor and setter methods work as expected.

Submission

This in-class lab is due at the end of this class. Do not upload your lab to BCIT's servers. When you are finished, show your instructor so he can sign your paper.

Checked by: _________________________

NOTE: keep this paper for your instructor to verify your grades later in the course.

NOTE: EVERY SINGLE STUDENT MUST SHOW THIS LAB AND GET HIS OR HER PAPER SIGNED....