The University of Melbourne

Department of Computing and Information Systems

COMP90041 Programming and Software Development

Lecturers: Dr Tilman Dingler and Dr Thuan Pham

Semester 1, 2020, Week 2

Workshop Instructions

Getting Started

The object of this week is for you to gain some familiarity with the mechanics of creating, editing, compiling and running Java programs using Oracle' Java compiler and virtual machine that will be the official Java platform for all the labs and projects throughout the semester.

It would be a good idea to work on these exercises in a pair with another student, so you can swap ideas. You are welcome to work in pairs or teams on all the unassessed workshop exercises, like those this week, but of course you must work alone on anything that will be submitted for assessment.

1. Setting up your environment

A wide range of Java compilers and Java Runtime Environments (JRE) are installed. Thus, you need to make sure that you are using the correct version. To do this, open a console window and type the following:

```
java -version
```

It should report back at least `Java 1.8.0`.

2. Using javac and java

Now you can try to compile and run a simple Java program. Download the Java source file `FirstProgram.java` from the subject's LMS page, and then copy it to a local folder. Open a console window, navigate to the folder where you saved the source file, and compile it using the following command:

```
javac FirstProgram.java
```

If all is well, `javac` will execute and finish with no messages. Then you can run the program by typing:

```
java FirstProgram
```

You should then see a hello and welcome message from the program appear on the console. If this happened then congratulations, you have just compiled and executed your first Java program!

3. Java IDEs

There are many Java IDEs (Integrated Programming Environments) available. We recommend Eclipse or NetBeans, but if you prefer another IDE, that's fine, too. Eclipse and NetBeans are installed in the labs; try both to see which you prefer. You can download them (for Windows, Mac, or Linux) for your own computer, and read through extensive tutorials at http://www.eclipse.org/ or http://netbeans.org/. They both support numerous programming languages and have numerous plugins, but we only need Java, and do not need any plugins, for this subject.

Whichever IDE you use, start up a new project, and put into it the same simple Java example as you used above. Run the program (via the "Run" tool), and observe the output. Really, the IDEs are just invoking the Java compiler and Java interpreter behind the scenes.

4. Workshop Exercises

These are just for practice, and will not be assessed.

1. Write a Java program that displays your name, address and telephone number each on a separate line.

2. Write a Java program that declares and initialises a String variable "`Hello, World!`" Print out the variable on a line by itself. Then, using the String methods discussed in Chapter 1, print out the same string in ALL UPPERCASE on a line by itself.

```
Hello, World!
HELLO, WORLD!
```

3. Modify the above program by just changing the initial value of the String variable "`So Long!`" and run it again. Now it should print out:

```
So Long!
SO LONG!
```