INTRODUCTION TO COMPUTER SCIENCE I
SETUP AND A FIRST PROGRAM
Friday, January 31, 2020

1 Getting Started

The following steps will get you started with this lab. You will be introduced to the programs and tools that you will need to complete the lab and project assignments. Please ask questions!

In this course we will use the IntelliJ IDEA Community Edition to write, compile, and run programs in Java, using the Java Software Development Kit (SDK). The SDK allows you to compile and run programs written in Java. IntelliJ is an Integrated Development Environment (IDE), which is a piece of software that allows you to develop, debug, and execute programs all in one environment. IntelliJ is capable of doing quite a lot, and we'll only see a small part of its functionality in this course. If you'd like to learn more, feel free to take a look at the documentation.

Before you begin writing and running programs, you will need to install the SDK and IntelliJ on your personal laptop (if you don't have a laptop, you're welcome to borrow one to use during lab).

1.1 Installing the Java SDK

The following are the steps to install the Java SDK:

1. In your browser, open the download page for the Java SE Development Kit 13. Accept the license agreement and download the installer for your platform (for Windows, download the .exe file, and for macOS, download the .dmg file).
2. In your browser, open the Overview of JDK installation page and click on the link for your platform.
3. Follow the instructions to install the SDK (note that you've already downloaded the installer at this point).
   - Windows users: you can ignore the section on "Installing the JDK Silently." After you've completed the installation, please follow the instructions in the section "Setting the PATH Environment Variable." You can ignore the rest of the instructions after that section.
   - macOS users: you can ignore the section on "Determining the Default JDK Version on macOS."

1.2 Installing IntelliJ IDEA

The following are the steps to install IntelliJ IDEA:

1. In your browser, open the download page for IntelliJ and select your platform near the top of the page.

2. Run the installer, following the steps in the wizard.

3. In your browser, open the Running IntelliJ IDEA for the first time page and follow the instructions. What you actually see on your computer might differ slightly from the instructions, but the instructions should be close enough to allow you to progress. When you get to the section on plugins, you can deactivate them all (we won't need any plugins for this class). Once you reach the welcome screen that gives you the option to create a new project, you're done with this part and are ready to move on to creating your first program!

2 Your First Programs

We'll start with the "Hello, world!" program, which you saw in class earlier this week.

1. In your browser, open the Creating your first Java application page and follow the instructions, starting from "Create a new Java project" (you can skip the "Install the JDK" section; we've already done that). In the "Project SDK" list, choose the one that starts with 13 if there are multiple options. Name your project `lab1` (the project will include more Java programs than just `HelloWorld`).

2. Continuing on the same page in your browser, follow the instructions in the section "Create a package and a class." Name your class just `HelloWorld` (i.e., skip the `com.example` prefix). After you click OK, you should see some starter code that IntelliJ has provided for you. This is there to help you get started—one of the big advantages of using an IDE for development!

3. Fill in the rest of the code for the `HelloWorld` program (you can copy the code from the handout for `First` that we saw in class, but give your program the name `HelloWorld` instead of `First` this time).

4. Build and run `HelloWorld`, per the instructions in the section "Build and run the application." You can skip the rest of the instructions on the page.

That's it! You've successfully written and run your first Java program! Now, we'll add another program to the `lab1` project, but this time, we'll import the code from the course web page.

1. Find the `src` directory (folder) under the project directory for your `lab1` project (this should be in the "Project" panel on the left side of IntelliJ).

2. Create a new class called `Conversation` the same way you did for `HelloWorld`. This time, delete all of the starter code that IntelliJ provides for you.

3. On the course web page, click on the `Conversation.java` link. Select everything in this file, copy it, and paste it into the `Conversation` file in the IntelliJ editor.

4. Run `Conversation` the same way you did for `HelloWorld`. You'll notice that this program is interactive: it waits for the user to type a response, then continues executing. You can type input to the program in the "Messages" window where IntelliJ prints the program output—just click in the window right below the program's output, and start typing.

3 Your Assignment

Your task is to modify `Conversation` to do something more interesting. Here are some ideas that you might want to try:

- Ask the user more questions. Use the existing code as an example for how to read the user's response from the keyboard.
- Declare some variables of type `int` or `double`, do some arithmetic, and print the results.
- Read numeric input from the keyboard. You can do this using the `keyboard.nextInt()` and `keyboard.nextDouble()` commands.
- Print some ASCII art.

After each change that you make to the program, be sure to run your program to test out your changes. It is in your best interest to run your code often as you make changes, as this will allow you to catch errors early. This may not seem like a big deal right now, but as you start writing programs that are more involved it will become increasingly important to write and test your code one piece at a time.

4 QuaRCS

This semester, COSC-111 will be participating in the QuaRCS study, led by Prof. Kate Follette in the Physics and Astronomy Department. From the QuaRCS web site:

> The Quantitative Reasoning for College Science study is a research study designed to investigate whether introductory science courses for non-majors can improve students' quantitative and numerical skills. The designers of the study hope that this will give us some insight into how to improve the usefulness of such courses.
>
> By completing the survey, which takes an average of 30 minutes but has no time limit, and clicking submit at the end, you are consenting to participate and to share your responses with its creators unless you e-mail the Principal Investigator and ask that your data be removed from consideration. Your participation is entirely voluntary, and you may withdraw at any time. There are no risks to you associated with your participation in this study.

To participate in the study, please fill out the survey: https://tinyurl.com/QuaRCSLt-Pre
You will be asked to enter your name on the survey; this will only be used (1) to match your responses at the start of the semester and then end of the semester (when you'll complete the

survey again), and (2) to let me know whether or not you have completed the survey (completing the survey is part of this lab assignment). I will not see your individual responses or results at any time.

5 Submission

Please submit your modified `Conversation.java` via Moodle.

This assignment is due on Thursday, February 6, 11:59 pm.