Indiana University
CSCI-C212: Introduction to Software Systems (S25) Lab 01
Joshua Crotts, Muazzam Siddiqui

IDE Setup and Basic Java

Objectives:

- Students will install and use an IDE, such as IntelliJ IDEA or Eclipse, as well as install a version of the JDK.
- Students can explain what is an IDE and why one should use it. They should understand the process of and motivation for using IDEs in Java.
- Students use the IDE to create our first program, wherein they learn to create a project, class, method, and statements.
- Finally, students learn how to submit files to the autograder.

What To Do:

Read the directions below and follow along with your TAs.

Part A: Listen to Your Lab TAs!

Normally, your labs aren't a full lecture. This time, however, we take things slow to get you acclimated with the IntelliJ programming environment. Please follow along with your lab TAs, who will introduce themselves, to install the IDE. They will also demo how to import the lab template into the IDE, as well as how to submit files to the autograder. **Pay attention to these demonstrations.** You'll be doing this all semester, so it's best to listen fully now, rather than not know what to do later on down the road.

You should also learn the names and emails of your lab instructors. They will share this on the first day of lab, but it is also publicly available on the course website. Do not email the course instructors several weeks into the semester with the question, "Who are my lab instructors?"

Part B: Lab Quizzes

Every lab, **including this one**, will begin with a 10 minute quiz on Canvas. (This time, however, you will have two quizzes: the first is about the syllabus and the second is about this week's content.)

These quizzes are worth 20% of the lab grade, with the coding component being worth the other 80%. You are allowed and encouraged to collaborate on the quiz as well. The quizzes are multiple-choice and are designed to test your understanding of minor details. What this entails is that they will be pedantic and you will probably not score perfectly every time, if ever. This is not something to be discouraged by, because they are intentionally tricky. If you have read the relevant textbook material, attended class, and done the necessary outside study, you will almost certainly get these fully correct. **You cannot retake a quiz. You cannot take quizzes early.**

Arriving late to the quiz means that you only get however much time you have to take it. Consequently, if you arrive after the quiz submission time, you will receive a zero.

If you have to attend another lab, please let ***ALL*** staff members, i.e., those in your assigned lab and those of the other, know via email (NOT DISCORD).

Part C: Paired Programming

Each and every lab will involve paired programming. So, get used to whoever you're sitting with, or sit next to someone you already know, because you'll be working with them (or those around you) all semester! Paired programming is for the benefit of all parties involved. By "paired programming," we do not mean "one person does all the work while the other person watches." This does nothing to the "watcher" and only serves to piss the "worker" off.

Here's what you should do: take turns with who types on the keyboard and who is talking through the problem. These roles are sometimes called "driver" and "navigator", respectively. Our variant of paired programming is not as strict as it otherwise might be; usually, paired programming is performed on exactly one computer. We will allow you to use multiple computers, but you should still be constantly talking and communicating with your group-mates. The staff will walk around to ensure that everyone is collaborating and contributing to the discussion and lab. **Refusing to work with a group will earn you a zero on both the quiz and the lab. Refusal to let other people participate in the discussion, i.e., "dominating the discussion," will also warrant a zero on the quiz and lab.**

Upon completion of a problem, switch roles. That is, whoever was first the navigator becomes the driver and vice-versa. Some people may think that this is a waste of time, and that they just want to complete the lab and go home. To these people we say: you signed up for this class and, more importantly, the **lab section you picked was at your discretion!** So, please take advantage of working with your peers directly. If you feel that your partner/group is not contributing their fair share of the work, please let your lab TAs know and they will either move you or try to facilitate discussion.

Finally, **do not complete the lab outside of class without prior permission, or get the lab document from another student.** If you complete the lab outside of class and show up expecting to get checked off, you will instead receive a zero for the quiz and the lab.

Part D: ASK QUESTIONS! COLLABORATE!

We cannot stress this enough: ask, ask, ask, ask questions. Do ***NOT*** simply sit confused. That is the absolute last thing that you should do. The second-to-last thing that you should do is use ChatGPT/AI to do the work for you, or simply Google the problem. Using ChatGPT is a violation of the academic integrity policy, and all incidents will be reported to the university. Moreover, none of these are beneficial to you or your group's learning. We, as the course staff, care about your learning, and if you are struggling, you need to let us know! As we have (and will repeatedly) expressed in class, please do not feel stupid or anxious about asking questions. If you have a question, there is almost certainly at least one other student with that exact same question, so you're doing them a service by asking.

At the same time, however, we expect collaboration and discussion. This means that, if the lab TAs are walking people through a problem, as a class, you should contribute meaningfully to the conversation and answer questions, even if you are unsure.

Part E: If you don't finish, don't fret!

As the section title suggests, if you don't finish every lab within the allotted lab time, it's not the end of the world. If you have consistently been on-task, working diligently with your partner/group, then this is generally enough for a very good score on the lab. It is also up to the discretion of the UIs to allow you to complete the lab after the deadline. If they (the UIs) do permit this, you are only allowed to submit up to 11:59PM on the day of your lab. For example, if your lab is at 1:50PM-3:45PM on Thursday, and your UIs let you finish the lab after the standard 3:45PM deadline, you must submit by 11:59PM on Thursday night.

Problem 1:

Design the `double celsiusToFahrenheit(double c)` method, which converts a given temperature from Celsius to Fahrenheit.

Problem 2:

Design the `int combineDigits(int a, int b)` method, which receives two `int` values between 0 and 9, and combines them into a single two-digit number.

Problem 3:

Design the `double pointDistance(double px, double py, double qx, double qy)` method, which receives four double values representing two Cartesian coordinates. The method should return the distance between these points.

Problem 4:

Design the `double crazyMath(double x)` method, which receives a value of x and computes the value of the following expression:

$$\sqrt{\left|\left(e^{-x} + \cos\frac{2}{x}\right) \cdot \left(\frac{\sin(\pi x - 2\pi) + 17x\pi}{\log_3 |x| \cdot \log_7 |x| \cdot \ln |x|}\right)\right|}$$

Below are some test cases. Hint: when testing this method, you may want to use the `delta` parameter of `assertEquals`!

```
crazyMath(0)      => NaN
crazyMath(1)      => Infinity
crazyMath(2)      => 21.52368973013284
crazyMath(3)      => 14.692493055407942
crazyMath(10)     => 9.574086130947974
crazyMath(100000) => 86.49768321282015
```