CS5004-5005-SV – Lab 13 – July 30, 2020
Maps in Java

This week's lab provides an opportunity to explore and gain practice with HashMap and/or TreeMap. Start by pulling up the lecture slides from 7/29/2020. Also grab a *copy* of the three .java files from the lecture, just for reference (GPS.java, Test.java, and TestMap.java).

0. Please create a 4-question ***quiz*** on the key ideas about *Maps* in Java, and when to use each type of map. Questions should be multiple choice or short answer. For example, you might ask what the default size of a HashMap is (16). In your breakout, ask your fellow students to take your test. If you don't get the same answers, discuss. Another example which be asking which things are interfaces versus abstract Classes versus concrete Classes. Your quiz may also provide a set of "boxes" that must be connect with arrows signifying whether the relationship is either *extends* or *implements*. In creating the quiz, it is fine to use the lecture slides, the Oracle language definition, or other Internet sources. Try taking your colleagues' quiz closed book, just for practice. Turn in your quiz including the answers. Any widely available file format is fine.

1. ***Requests: Small Lab Project.*** Several students requested a "small lab project" that might take more than a single lab session to complete, require team collaboration, and so on. My main concern is not to overload everyone as "Final Exam" and "Homeworks" stress reaches a crescendo. Hence, I have tried to come up with an assignment that meets the following criteria: ***(a)*** possible to complete in ~2 hours per week over at most 2 weeks. Please don't get sidetracked by it! ***(b)*** Since not everyone is familiar with creating a graphical user interface (GUI) with Javax Swing, nor with design patterns such as Model-View-Controller (MVC), I recommend a text-based game. ***(c)*** At least one of **HashMap**, **LinkedHashMap**, or **TreeMap** should be necessary. ***(d)*** You should work with a partner ("pair programming"). ***(e)*** Preparing for Finals takes priority; but, time permitting, each team is encouraged to do a 1-minute walk-through of *one* Class in their project during breakouts at next week's Lab.

   Here is an *example idea* that can meet this rubric. *Start* on a **text-based**

Adventure Game written in Java. Use an Enum declaration to specify directions for exiting a room or "chamber" in the adventure. The combination of a room name (String) and an Enum (such as NORTH) should suffice to look up the name of a next room to visit. The room name can serve as a Key to provide various information about the room (what exit directions are available, whether there are objects or "bad actors" in the room, etc.). In case you aren't familiar with this genre of games, here is an example to play with. This version happens to be written in Python; what matters is seeing how Maps can be invaluable in this type of project. You should do *your* work in Java.

https://learningtech.org/adventure/

Do not try to implement any features ***except*** room-to-room navigation, at least not until ***after*** the end of summer term! Also, do not worry about making it run within a web page!

You are ***free to choose a different example*** -- but please limit your efforts, including debating project alternatives, to a maximum of 3-4 hours over this week ***and*** next! Completing all the homeworks and studying for Finals is much more important. Beautiful, finished code is not expected, just enough to exercise some of the ideas about Maps in Java.

Submit your answers to above items to Canvas for CS5005 *(due next Thursday by midnight). Only your .java files and any associated .txt files for the above need to be uploaded. (Often you can just add comments within your code files as needed to identify the problems and to answer questions about the code or clarify your approach.) Timely submission encouraged, but late is better than never. Then work on any unfinished labs or CS5004 homework assignments. Remember to submit* ***homeworks to Canvas-for-CS5004*** *and* ***labs to Canvas-for-CS5005****.*