CS100J, Spring 2001 Project 6
Due Thursday 5/3

1. Objective

Completing all tasks in this assignment will help you:

- Learn about inheritance and polymorphism.
- Learn about advanced programming topics: command-line arguments, I/O, exceptions, and **Vector**s.
- Apply MATLAB for developing qualitative understanding of a model from a collection of data.
- Use previously taught concepts to understand how to build a larger, more comprehensive program.

First skim, and then, carefully read the *entire* assignment before starting any tasks!

2. Tasks

- *Thoroughly* inspect all Java files posted in the Project 6 portion of the website. This is a different kind of project– you will be working on understanding how a larger program is assembled.
- Draw the inheritance hierarchy for all classes used in the program. If you prefer, you may skip **System**, **Vector**, **Math**, and any other built-in class except for **Object**. Title your drawing "Simulation Hierarchy."
- Finishing commenting the Java files. The "bigger" ones are already done. Use Savtich's style of pre- and post-conditions.
- Complete the blanks in **Dishroom_blanks.java** such that the entire program will compile and run. The blanks are in the **extractContents** method. Hint: You will likely need to look things up in Savitch or another book.
- Inspect the output that the program generates. On a sheet of paper, explain why or why not the output seems "reasonable." You are required to test the output! On the same sheet, explain what technique/approach you took and what you discovered. Hint: Use numbers that you can trace.
- Run your completed Java program for cases of 1 to 10 **Worker**s with 100 tests for each. Note that you will need to use a data file to supply this input.
- Create a MATLAB program that will import the data generated by the Java program. The MATLAB program will compute and output the average amount of **Tray**s and **Item**s a group of **Worker**s can extract. The program will also generate plots that help show the distribution of the amount of **Item**s and **Tray**s the **Worker**s could extract. To make things easier, add the amount of **Tray**s to the amount of **Item**s. Hint: You should generate plots of amount of **Item**s versus amount of **Worker**s. Another plot could be the amount of Items versus the amount of **Worker**s versus the number of tests of the model. See **hist** and **plot3** for some ideas.

Note that some of these tasks might seem hard since we *just* started MATLAB. DIS will be supplying special "hint" files to help you along with the MATLAB process.

3. Bonus Tasks

Each of the following tasks is worth 1–5 points for the assignment.

- create and use an inheritance hierarchy for the **Worker**s (see top of **Worker.java** for an approach)
- redo all comments to suit Javadoc requirements (see Savitch)
- write your own simulation that improves upon the "retrievability" model: you must explain and justify your approach. This program would be in addition to the required portion of this project.

4. Submitting Your Work

Submit a title sheet, table of contents, all programs, discussion, and output indicated in this assignment. Follow the submission guidelines stated on the Projects page for CS100J. Number all pages to help the graders find each portion of the assignment.