**CIS 640**

**Homework 4**

**White Box Testing**

For this homework, you will be testing a simple vending machine Java application. A state diagram of the machine can be found below. Main.java is the controller for this program, StockItem.java is a class to represent an item in the vending machine, and VendingMachine.java contains the majority of the functionality of the machine itself. The vending machine can be loaded using a text file. The file is formed with 1 item per line. Each line is formatted as **item\_name,cost,quantity**. A sample file is included, but you should not limit your testing to using only this file! You are not required to draw the control flow graph for this project; however, you should utilize the strategies you learned doing control flow and data flow testing.

For the assignment, you are to write both unit tests and integration tests with Junit5 to achieve as close to 100% branch and statement coverage as possible. Coverage can be measured using built in features of Eclipse or IntelliJ, or you may download and use [JaCoCo](https://www.eclemma.org/jacoco/index.html). You should have a **minimum of 10 unit tests** and **5 integration tests**. This won’t necessarily obtain the desired coverage, this is just a lower bound. Your unit test cases and integration test cases should be in separate Java files and be named clearly to indicate what type of tests the files contain (individual tests should also be named to reflect this). Points will be awarded based on the coverage you achieve, as well as the quality of your tests.

You should be turning in a zip file that contains all Java files you created that contain your tests, a document (this may be some screen shots) that indicates your coverage, and a document that lists your tests with a short summary, if that test passed, and if it failed (there are some bugs in the program), explain the bug.

