

SYSC2004 – Project Report  
Milestone 3  
Sebastien Marleau  
SN: 101155551

### **Change log:**

- ➔ Created Test classes for both ShoppingCart and StoreManager
- ➔ Did not add any additional exceptions or errors, as the bugs from the user input and other parts of the code were already handled with conditional logic.
- ➔ Refracted all UI elements from StoreManager into StoreView, and added accessing methods in StoreManager.
  - I thought it would be better for StoreManager to have no UI components in it
  - This could also be a preparation towards the creation of the GUI

### **Questions**

#### **1. What were the testing methods/strategies that you used for this milestone? Be detailed. Use the testing terminology presented in lectures.**

The testing was done with a library called JUnit. Using assertions, it was possible to make sure the code was behaving the way it was supposed to.

I tried to test most of the edge cases, such as when an integer argument is negative, etc.

#### **2. Were there some things you were unable to test with JUnit? What were they, and why were you not able to?(Think about the levels of testing.)**

I was able to test almost all the code in the two individual classes we were asked to make tests for (StoreManager and ShoppingCart). I was not really able to test the interaction between these modules too much though.

#### **3. With respect to Question 2, should these parts of the code be tested? Should every inch of code be tested in general?**

It depends how foolproof you want the code to be. As it takes time, effort, and money to get people to write test cases, usually it is not worth it to test every inch of the code. Sometimes though, with really important pieces of code, it may be necessary to do so to make sure it crashes the least amount possible.

Furthermore, it is often not necessary to test some parts of the code, such as getters, setters, and functions that just return a function call from another object. These are either so simple there couldn't really be a mistake, or call a function from another object, so the test case for that function should occur on the test bench of that object.