Eric Morton

CS 121

May 2, 2015

Set-Game Write Up

This past month, I was given the opportunity to create a game called Set through Java. For the game Set, there are four categories: Shape, Quantity, Color, and Shading. In order to get a set an individual needs to have each of the three cards to be one of the four categories but they all need to be different. At the beginning of this project I was afraid that nothing was going to work right, but I took a breath and took each part as it came.

Starting the project with the Card class was the easiest part for me. I was able to put certain pieces together and use the notes that I took in class to use. The Card class was as trial and error period for me. I knew that we have to initialize instance variables, but one that thirty-second process was done; I was at a stand still. Thus, I began to code while following the instructions. Finishing what I thought was a finished piece of compiled code; I realized that I had to test the code that I had just written. When I pressed test, I had many errors which means I had to go back and fix what I thought was correct. After a trial and error period I finally got all my tests to pass, and with Professor Sommers tests passing as well ensured me that this was going to be a good project. With the Card class done, it was now time to move on to the Deck class.

The Deck class required the class was designed that cards are added to the cards ArrayList at creation but never removed. In order to keep track of the cards that have been removed from the deck, we have to use the integer value nextCardIndex. Going into this I was prepared for what was to come. After getting a one hundred on the first part of the project, the Card class, I knew that this was going to be an “Ace in the hole”. Writing the Deck class was quite challenging, but not to the point where I could not handle it. The same technique that I used for the Card class I used here, but the only difference for this part was that I asked for assistance from my peers and my teacher’s assistant. With a complied code, I ran only one test and this is where I made one of my many mistakes. As Professor Sommer ran his tests I received no credit. Upset, I went back to my code and had one of my peers edit it. There were every small errors in my code, such as having a double equals when there should have been one equals, forgetting parenthesis at the end of certain methods. Still upset with my self, I knew I had to pick up the slack on the next part: Table.

The Table class may have been the most tedious, annoying, frustrating part of the entire project. For this part of the project, drawing the picture is essential for the nodes for our linked list. For this part, we had to add cards to the table and remove a set from the table if one is there. This seemed easy at first, but there was a lot that had to go with it. I had to actually play the game set to understand what was going on. Numerous times I reached a part where I did not know what to type thus I went to first my Professor then my peers. After hours of working on this part of the code, I ended it with a solid B. This was good enough for me, my mind was now on the last part of the project: Game.

For the Game part of the project I was on my toes. I knew how to write the code, and how to go about executing it. Game required up to actually play the game Set. With the list of all the possibilities given to the class, I took one step at a time and checked behind myself. I found myself making less for loops for this part of the project than I have done for the Table class. This was the easiest which made it more fun for me to write this part of the code. Getting all my tests to pass and my Monte Carlo simulation to work brought me tears of joy because I knew that I have completed the project.

I have learned quite a bunch after finishing this project. The first is that Computer Science is not for me anymore. Coming into Moravian College I was set on Computer Science as a major but after taking 120 and 121 I realized that coding is not for me. The second thing that I learned is that I need to re-read my coding as if it was a paper for another class. It was always grounded in me to check my work when I am done, but I did not apply it to my coding skills. If I had to do this project again, I would come to my Professor more often if I have questions, or rather work on it a week before instead of waiting until the last two days to do the project. All questions cannot get answered in a two-day span.

On future projects, I will use my time wisely and efficiently to make sure that I get all my questions answered before I turn my project in. Secondly I will make sure that before I turn my project in, I review it to be sure that all of it is correct. I cannot afford to get points taken off for silly mistakes. All in all, even though the project had its ups and downs, I did enjoyed doing the project. There is a great sigh of relief to know that a challenge has now been over come.