Jamie Loebe
CS 162
Lab 3 Reflection

   I began working on Lab 3 with a good feeling towards how it would work and feeling like I had a pretty good handle on it. My original design was to work through the program from top to bottom and just start out with a basic idea. I wanted to create the Die class first and be able to have my random number generator and Roll() function working. I felt that as soon as I had that working, I would have a good start to the lab itself. As soon as I got started and was able to get the roll function working, I continued towards the LoadedDie class and inheritance. I was a little tripped up at times and it took a little while for me to completely understand inheritance and virtual functions, which I actually had to circle back to. At first, I had attempted to code the program without virtual function but ended up working my way back through my code to account for virtual functions as I was encountering problems without it. My program for a period was allowing the user to input the type of die, rounds to play and number of sides on the die. However, it was crashing immediately after the input was received. I ended up working through each line of the code, working through the book and researching online. Though many attempts of trial and error, I ended up using virtual function for my Roll function and was able to have the code work! However, I ran into another issue. I had up until this point been coding on Xcode on my mac. Once I transported my code to flip however, I ran into the same issue. Input was received, and the program crashed, outputting garbage values to points and rounds played. Once again, I went back through my code, line by line, and found that I had actually been inputting the wrong variable in for points and the roll function. Because of this, my program was crashing due to not knowing what to do with the wrong variable. I went back through, made the quick change to the points variable and roll function and my program was a success. I also ended up going back and remembering to deallocate my dynamic memory as I forgot about that the first time through and ran valgrind to check and confirm that there were no memory leaks. Throughout this program I. relied heavily on the textbook as well as my own online research fkr a better understanding of inheritance and virtual functions.

Test Case

Test Case	Input Values	Functions	Expected Outcomes	Observed Outcomes
Input not 1 nor 2	4	If (beginGame)….	Terminates game	Terminates/outputs Rounds played =0
Input = 1	1	If (beginGame)….	Begins game	Begins game, asks for input
typeOfDie = no	no	If(typeOfDie.compare) ….	Creates unbiased normal die for player	Creates unbiased normal die (Not loaded) for player.