Lab 3

Goals-

Adapt code you did not write according to your design

Develop a plan to test your program against requirements and verify the results

You will implement the program designed in Lab 2. You will receive the lab 1 files from another student. You need to do nothing as the TAs will distribute the code files. You will then modify the lab 1 files to have LoadedDie inherit from Die, in accordance with your design from Lab 2.

Your program will use: the loadedDie class received from a classmate and modified by you, your die class which will now be the parent of loadedDie, and you will use the test program received from the same classmate.

You will need to document the changes required for the code and the design. If you encounter problems please describe it and how you intend to correct it.

NOTE: Just throwing out the code you receive is NOT an option.

Test the finished program. Do the results make sense? That is, when both are using the same type of dice are the results close to 50-50? If one is using loaded dice are the results skewed in their favor? If not, what is the problem?

Modular Grading

We are using modular grading. Each lab will be divided into specific modules. Each module will be graded pass/fail. It either works properly or it does not. 10% of every lab or assignment grade is style/comments or other elements of self-documenting code and clarity. Remember the labs are worth 10 points total.

Programming style- 1 point

Create the modified derived class source and header files - 3 points

Use your test program from Lab 1 to verify the results- 2 points

Test using different combinations of Die and LoadedDie- 1 point

Design changes and analysis of results - 2 points

Use your makefile from Lab 1 to compile your program- 1 point

Note: This lab requires a second short pdf document with the analysis and results.