Assignment #5 – Inheritance

Date assigned: Monday, April 12, 2010
Date due: Monday, April 19, 2010
Points: 25

For this assignment, you are to implement the following classes:

1) class Die in Die.h & Die.cpp - class Die is a generic Die class that will be inherited from other classes

2) class DieSixSided in DieSixSided.h & DieSixSided.cpp - class DieSixSided is a six-side die class that can roll values between 1 and 6 inclusive. This class inherits from class Die.

NOTE: So why even have a Die class? Well we might create other die as well like a DieTenSided for a game. Having a Die base class allows us to reuse code.

You must first decide what all Die have in common. The variables and behavior that all Die have in common will be placed in the Die class and inherited by DieSixSided.

Using inheritance, you are to implement the class DieSixSided. The DieSixSided class is to have the ability to set, get, roll, and display appropriately. The method roll will use your previously implemented Random class to randomly roll the Die. Just copy the Random code into your project. The display method is to simply output the Die value.

You are to create one other class for testing class DieSixSided as follows:

1) class `DieSixSidedTest`

This class is to create a single DieSixSided object using the randomSeed 1234 and roll the Die 100,000 times. Use an array with six elements to collect the number of times a 1, 2, 3, 4, 5, and 6 were rolled. Output your results exactly as follows:

```
*******************************************
*    MultipleDieSixSidedRolls Execution    *
*******************************************

Side 1 occurred xxxxx times
Side 2 occurred xxxxx times
Side 3 occurred xxxxx times
Side 4 occurred xxxxx times
Side 5 occurred xxxxx times
Side 6 occurred xxxxx times
```

The use of well-defined classes above will result in a main that simply creates a DieSixSidedTest object. Calling the appropriate method will display the desired results.

Goals for Assignment #5:

1. Write more complicated C++ classes using inheritance
2. Continue to debug and test our classes one method at a time
3. Review and use the basic concepts of OOP which are composition and inheritance

Specifics:

1. Save your project as 05PUNetID fully and correctly documented.

2. Your code is to be written using Visual Studio 2008 and placed in the CS250 Drop Box by 9:15am on the day in which the assignment is due. A stapled hard copy must be placed on the instructor's desk before 9:15am on the day the assignment is due for the entire project to be considered on time. Remember, print main.cpp, and the .h/.cpp related files for each class.

3. If you come to me with a question regarding your solution and I need to look at your code, I will either grab your code off of Turing or a jump drive. I will not look at your code on your computer as it just takes me too long to get at the problem. Further, I want you to bring in your textbook and lecture notes in case I want you to look up something. Remember, I'm not just a tell you the answer guy.

4. If you want help with a compiler error, you must be able to tell me exactly what statement you put in your code that caused the error and be able to isolate the error. If you have typed in a bunch of code and have not tested your code as you've gone along, I'm not going to help you sort out the mess. You've been warned!!