**CS 4322 – Coding Assignment 5**

Brandon Corbett and Zackery Weeks!!

**Problem**

(75 points) Design a system to model one-dimensional, real-valued functions using the decorator pattern as discussed in class. Your design should handle functions of any complexity that utilize: arithmetic operations, exp, square, square root, power, sin. All classes must be immutable.

For additional points add some of these capabilities:

1. X(5 points) Round decorator that rounds a value to a specified (in constructor) number of decimal places.
2. X(5 points) Combination decorator, . An exception should be thrown if *x* is invalid. See:
3. (5 points) A piecewise decorator composed of two pieces. Note: you will also need a way to specify the domain for each piece. You should probably use a Boolean to indicate if a domain boundary is exclusive or inclusive, *i.e. .*
4. X(5 points) Max decorator that operates on two functions
5. X(5 points) Max decorator that operates on any number of functions. If doing this requirement, you should have one decorator that works for both this requirement and requirement 4, using different constructors to handle the two cases.

**Requirements:**

1. Provide a write-up explaining:
2. Your design which includes a class diagram(s).
3. What works and what doesn’t and which additional requirements you implemented.
4. Time Log(s)

We started out by talking about what the project was asking and tried to get a good idea of how the decorator design worked. When we started writing our assignment we were referencing the book heavily. We finished writing our Function class which is an interface for our constant and XValue classes. When creating our MathDecorator we wrote it using the book as a reference which we eventually realized was wrong after talking to you. We then had to go back and change our MathDecorator to hold the object references in an arraylist rather than the being held in each individual wrapper class. We created all the decorators that were asked for except the piecewise decorator, because we didn’t find a way to make this one work. In our tester, we have created two tests for each decorator. We tried to include specifications for each that we felt tested the code in different ways. In our second test for combinations you will notice that it has been commented out. This is because the test will fail since the second value is larger than the first value. You can uncomment it to show that it will throw an InvalidArguementException when ran. We have included a class diagram as ClassDiagram.ucls.

1. Write the code including a tester

**Final Deliverables**

Zip (or Rar) the items described above into a file named: *ca5\_lastName1\_lastName2.zip* and submit on Blazeview. Please only submit once, under one person’s account if working in a group.

**Time Log**

If working in a group, submit a single timelog ONLY if you worked face-to-face for the entire assignment; Otherwise submit one for each member (copy table below).

* **Delete empty rows, add if needed.**
* **Put the total time at bottom.**

|  |  |
| --- | --- |
| **Time (hrs)** | **Task** |
| 2 | Talking it out and deciding where and how to start. |
| 3 | Using the book and coding the Function class with Constant and xValue classes |
| 2 | Wrote the MathDecorator class and started on the wrappers |
| 3 | Realized the book was wrong… rewrote the wrappers that were finished and started on others |
| 4ish | Finished the wrappers (except piecewise) classes and wrote tester |
| **14** | **Total Time (hrs)** |