Student ID_____Student Name_____

Advanced Software Development DE Midterm Exam
May 7, 2016

PRIVATE AND CONFIDENTIAL

1. **Allotted exam duration is 2 hours.**
2. **Closed book/notes.**
3. **No personal items including electronic devices (cell phones, computers, calculators, PDAs).**
4. **Cell phones must be turned in to your proctor before beginning exam.**
5. **No additional papers are allowed. Sufficient blank paper is included in the exam packet.**
6. **Exams are copyrighted and may not be copied or transferred.**
7. **Restroom and other personal breaks are not permitted.**
8. **Total exam including questions and scratch paper must be returned to the proctor.**

7 blank pages are provided for writing the solutions and/or scratch paper. All 7 pages must be handed in with the exam

BE VERY CAREFUL WITH THE GIVEN 2 HOURS AND USE YOUR TIME WISELY. THE ALLOTED TIME IS GIVEN FOR EVERY QUESTION.

Write your name and student id at the top of this page.

Question 1 [15 points] {15 minutes}
Below you find a list of design problems that can be solved by applying one of the patterns we used.
Give the name of the pattern you would use to solve the given problem. Only write down the name of the pattern

Design problem	Name of the design pattern that solves this problem
We want to be able to sort a list in different ways like using bubblesort, insertionsort or even your own sorting algorithm	
We want to be able to record and play back all actions we do when we play a game	
We want to store the content of a XML file in memory and then do operations on this XML content like adding a new element, or changing an attribute	
We want to draw certain shapes in a paint application. We first have to select which shape we want to draw (circle, line, etc). Then we click the mouse on a canvas at position 1, followed by moving the mouse to position 2 while holding the mouse down, and then releasing the mouse at position 2. The application should now draw the selected shape using the 2 positions.	
We want to implement the model-view-controller pattern that separates the model (data we want to show) from the view(user interface that shows the actual data). When the model changes, the views have to be updated. You can have as many views as you want.	
We receive different kind of orders from all of our clients. Every order has its own format and we have many clients. So whenever we receive an order, we first have to know who is the client that has sent this order, end then we know how to process it. We expect to add many new clients in the future that will sent their orders in their own format. It should be easy to add many new clients.	
We want to loop over a collection without being dependent on the structure of that collection.	
We want clients of a component to be independent of the internals of that component	

Question 2 [25 points] {25 minutes}

Suppose we have the following Stock class:

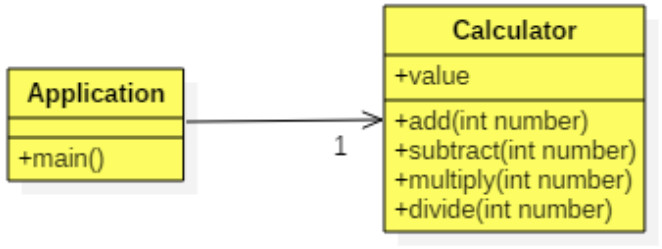
```
public class Stock {
      private String symbol;
      private double price;

      //getter and setter methods are not shown
}
```

We need to add the following functionality:
Every time when the price of the Stock changes, other objects (EmailSenders, SMSSenders, WebServiceCallers) want to know about this change. The **number** of objects and the **type** of objects that need to know when the stock price changes should be variable and is not known upfront.

a. Draw a **class diagram** of your design. Make sure you apply the principles and best practices we studied in this course. Make sure your class diagram contains all the important information to communicate your design.
b. Draw a **sequence diagram** that shows what happens when the price of the stock changes.

Question 3 [25 points] {30 minutes}

Given is the following simple Calculator application.:



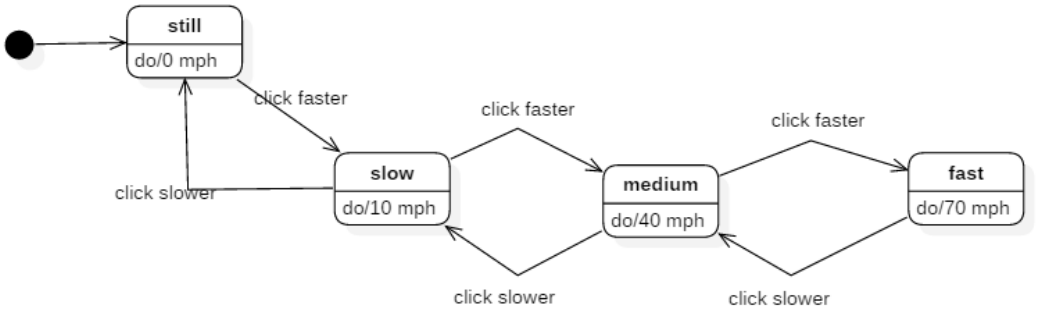
You need to modify the given application so that the application also support undo/redo functionality.

a. Draw a **class diagram** of your new design of the calculator application. Make sure you apply the principles and best practices we studied in this course. Make sure your class diagram contains all the important information to communicate your design.
b. Draw a **sequence diagram** showing the following sequence:
    1. First we add 4
    2. Then we subtract 6
    3. Then we want to undo the last action.

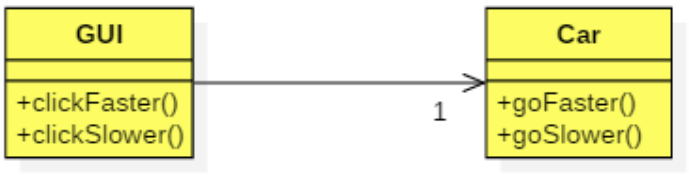
Question 4 [30 points] {40 minutes}

Suppose you have to write the software for a car racing app that runs on a mobile phone. You can use the motion detector of the phone for steering the car to the left or the right. But to control the speed of the car we want to use 2 buttons that are shown on the display of the phone. There is a button with the text "**faster**" and a button with the text "**slower**"

When the game starts the car is standing still, so 0 miles per hour (mph). The control of the speed of the car should be as follows:



So whenever you click the faster button, the speed will go up to the next level and when we click the slower button, the speed will go down.
A simple design for this application will be as follows:



But in this simple design it is not easy to add new levels like ultra-fast or ultra-slow. Make a design that make it easy to add new levels.

c. Draw a **class diagram** of your design. Make sure you apply the principles and best practices we studied in this course. Make sure your class diagram contains all the important information to communicate your design.
d. Draw a **sequence diagram** showing the following sequence:
    1. First we click the faster button
    2. Then we click the faster button again
    3. Then we click the slower button.

Question 5 [5 points] {10 minutes}

Describe how we can relate the **Mediator** pattern to one or more principles of SCI. Your answer should be about half a page, but should not exceed one page (handwritten). The number of points you get for this questions depends how well you explain the relationship between the **Mediator** pattern and one or more principles of SCI.