

Due: October 20 (**Saturday**), 11:59pm

Cutoff: **No late submission** is accepted.

Points allocated: 80 points (50 points for program, 30 points for documentation)

Note: the grade of this homework is determined by the implementation of polymorphism and design patterns, without using which students may not get any credit.

Using Hangman, Baseball, and TicTacToe games developed as Homeworks 1, 2, and 3, write a program for an arcade game as follows:

- Run a demo at `cs.uco.edu` to get an idea of how Homework4 should work:
 username: oop, password: fall2012
 java -jar hw4demo.jar
- Note that the demo does not include the actual play of each game, but shows the overall flow of the program.
- When the user chooses “quit” menu, the statistics of the game is displayed at the ending screen. The statistics includes (refer to the demo program):
 - the history of all games played (name and score)
 - for each type of game, the game count and average.
- Students are allowed to design their own scoring systems which reasonably reflects the skills of the game player

Program Requirements

- Each of three game programs must work without any error.
- Use object oriented design concept **in two key areas**.
 - Implement **polymorphism**.
 - Implement **at least two design patterns** we discussed during the lecture.
 - **Note** that you are required to “implement” a design pattern. This means that simply using a design pattern from the Java library does not fulfill this requirement (e.g, use of the Iterator or ArrayList).
- **Homework4** is the name of the main class.
- All classes must be **public** unless you can justify otherwise. This implies you cannot put more than one class in the same file.

Documentation Requirements

- Draw a UML diagram using conventions used in the lecture notes.
 - Do not simply put all methods and instance variables.
 - Instead, only put methods and instance variables required in the essay of polymorphism and design pattern.
 - The diagram must be reasonably readable when it is printed on the paper.
- Write an essay as to how you implemented polymorphism and two design patterns in the assignment. (minimum one page in **single** space, font size 12 pt)
 - When explaining polymorphism you implemented, you must use clear references to the UML diagram.
 - When explaining the design patterns you implemented, you must explain it in term of “participants” and with reference to the UML diagram.

- Convert the document into one single PDF file. (You may use PrimoPDF - a free pdf converter)

Turn-in Procedure

- Create a folder named “homework4”, and put all source files (java) and the document (pdf) into the folder.
- Compress the folder (homework4.zip), and submit it via D2L.