

Homework #5

TA in charge: ChiLin Cheng

RELEASE DATE: 05/05/2009

DUE DATE: 05/19/2009, 14:20

As directed below, you need to upload your submission file to the designated place on the course website.

Any form of cheating, lying, or plagiarism will not be tolerated. Students can get zero scores and/or fail the class and/or be kicked out of school and/or receive other punishments for those kinds of misconducts. Discussions on course materials and homework solutions are encouraged. But you should write the final solutions alone and understand them fully. Books, notes, and Internet resources can be consulted, but not copied from.

Since everyone needs to write the final solutions alone, there is absolutely no need to lend your homework solutions and/or source codes to your classmates at any time. In order to maximize the level of fairness in this class, lending and borrowing homework solutions are both regarded as dishonest behaviors and will be punished according to the honesty policy.

Both English and Traditional Chinese are allowed for writing any part of your homework (if the compiler recognizes Traditional Chinese, of course). We do not accept any other languages.

1 Description

The `POOPet` is ready to breed, and fight! You have lots of flexibility to design a fight game between `POOPets` this time. A fight game happens at an `POOArena`, with two `POOPet` objects inside. Each `POOPet` comes with at least three properties, its health point (HP), its agility value (AGI), and its name, and with at least two actions, joining an arena, and acting. Its acting can include, but is not limited to, moving, dodging, attacking, etc.. The goal is to decrease the opponent's HP as soon as possible.

The job of `POOArena` is much like the `Table` in your homework 3. It provides a basic drawing utility to let us see what's going on in the arena, and facilitates communication between the fighting pets, such as how HP gets decreased.

You are asked to implement and demonstrate a prototype of the fight game. As it is only a prototype and probably will be extended by the programming team later, both `POOPet` and `POOArena` need to be **abstract**. Nevertheless, you can resort to polymorphism and use subclasses of `POOPet` and subclasses of `POOArena` to make things work. You need to implement at least two kinds of (i.e. two different subclasses of) `POOPet`, and one kind of `POOArena`.

We have provided the source code to a `POOFight` class to let you know about a sample fight game. We will use the class to test your program—so be sure to implement all the necessary methods. If your code deserves a better demo program, feel free to write it by yourself and let us know.

2 Requirements

- Implement your `P00(PetKind)` and `P00(ArenaKind)` classes, and any additional classes that you need.
- Write a short report with at **most** two A4 pages that contains the following items:
 - (1) your name and school ID
 - (2) the relations between the classes that you design
 - (3) the advantages of your design
 - (4) the disadvantages of your design
 - (5) any part that you implemented that is worth getting “bonus” points

You should submit your report in **PDF** format. See <http://jsc.cc.ntu.edu.tw/ntucc/pcroom/manual/Word2Pdf.htm> for some possible instructions for converting from Word to PDF.

3 Special Notes

- Readability of your source code would be worth 10 points out of 100 this time. The source code would be read by all the three TAs, each giving points based on the following qualitative measure:

- 10 very readable
- 8 readable
- 6 mostly readable, but with some unreadable parts
- 4 mostly unreadable, but with some readable parts
- 2 unreadable
- 0 very unreadable

Your score in this part would be the average number of points from the three TAs rounded to the nearest integer.

- 20 bonus (a.k.a. extra) points will be allocated to award creativity—basically how interesting your game is.
- Some sample results from the TA's program can be found on the course webpage.

4 Submission File

Please upload a single ZIP encrypted file to CEIBA. The zip file should be like `b86506054.zip`, where the file name should be changed to your own school ID. The ZIP file should contain the following items:

- `P00(PetKind).java` (at least two different kinds)
- `P00(ArenaKind).java` (at least one)
- `*.java`, which represent any other classes that you implemented
- `README` (optional), which instructs the TA to compile your files
- **Your report file in PDF format**
- **YourDemo.bat or YourDemo.sh following the sample Demo.bat**