Homework #3
RELEASE DATE: 03/29/2013
DUE DATE: 04/11/2013, noon

As directed below, you need to submit your code 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 get negative 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.

You need to write your homework report in English. For programming, we only allow Java.

This homework set comes with 200 *points and* 25 *bonus points. In general, every homework set of ours would come with a full credit of* 200 *points.*

1 Description

`POOArticle` is a key class that helps users post their articles on the POO BBS. It should contain the following member variables to store an article:

  (1) ID: a unique **3-digit** integer for differentiating between two or more articles

  (2) title: a string for storing the title of the article

  (3) author: a string for storing the author name of the article

  (4) content: a string for storing the content of the article

  (5) evaluation count: an integer for counting the number of pushes or boos

  (6) evaluation messages: a fixed-size array of strings, each of which stores one line of evaluation message from the user
     –the size of the array is defined as a constant (static final) member MAXEVAL of the class.

Moreover, as in the usual BBS system, users can do some actions on the articles including "boo", "push", "arrow", "list", "show". Those actions are implemented as the methods of the class, and work as follows:

  (1) a "push" method that increases the evaluation count by 1, and adds a line to the evaluation messages

  (2) a "boo" method that decreases the evaluation count by 1, and adds a line to the evaluation messages

  (3) an "arrow" method that does not change the evaluation count, and adds a line to the evaluation messages

  (4) a "show" method that shows all the article information, the content, and the evaluation messages

  (5) a "list" method that shows only the evaluation count, the ID, the title, and the author

Note that your push/boo/arrow methods should take care of the case where the message array is full, and cannot insert messages anymore.
Last but not the least, there should be a constructor, which takes the title, author, and content strings as arguments, and assigns a unique 3-digit integer ID to each `POOArticle` instance.

Then, we will need a `POOBoard` class that implements the boards on the BBS. A board is a collection of at most 1024 consecutive `POOArticle` instances. With the `POOArticle` class, the `POOBoard` class should implement the following methods:

```
public POOBoard(String name){
  //create a board with the name
}
public void add(POOArticle article){
  //append the article to the board
}
public void del(int pos){
  //delete the article at position pos
}
public void move(int src, int dest){
  //move the article at position src to position dest
}
public int length(){
  //get the current number of articles in the board
}
public void show(){
  //show the article titles of the board
}
```

In addition, there should also be a `POODirectory` class that can be used to implement a directory like "My Favorite." The directory is a collection of at most 1024 directories, boards, or splitting lines. The `POODirectory` class should implement the following methods:

```
public POODirectory(String name){
  //create a directory with the name
}
public void add(POOBoard board){
  //append the board to the directory
}
public void add(POODirectory dir){
  //append the directory to the directory
}
public void add_split(){
  //append a splitting line to the directory
}
public void del(int pos){
  //delete the board/directory/splitting line at position pos
}
public void move(int src, int dest){
  //move the board/directory/splitting line at position src to position dest
}
public int length(){
  //get the current number of items in the directory
}
public void show(){
  //show the board/directory titles and splitting lines of the directory
}
```

There are several designs that can achieve the functionality above. Your task, as an experienced programmer, is to come up of your favorite design about the relations between `POOArticle`, `POOBoard`, `POODirectory`, and any other classes that you want to implement. Then, realize your design, write a small demo program to help your poor TAs, and defend your design with a written report. How do you want those classes to interact with each other? What are the advantages of this design? What are the disadvantages? Your arguments can be about implementation time, code reusing, hierarchy meanings, future maintenance, etc.. Please use your creativity and come up with an excellent report!

2   Requirements

- Implement the `POOArticle`, `POOBoard` and `POODirectory` classes, and any additional classes that you need for your design.

- Do some version control for maintaining your code (see Submission File), which would be worth 20 points out of 200 this time. That is, your grading TA would give you points based on how/whether you try to do version control.

- Write a short report with at **most** four 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) how you "maintain" your code in your repository (see Submission File)

  (6) 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   Submission File

We will guide you to some GIT service to allow you to learn to manage your codes for the homework. The exact details of the submission will be provided later. Your GIT repository should contain the following items:

- `src/POOArticle.java`

- `src/POOBoard.java`

- `src/POODirectory.java`

- `src/*.java`, which represent any other classes that you implemented, including the demo program

- `Makefile`, where the TAs can use `make` on CSIE R217 linux machines to compile your code, and then `make run` to execute your demo program

- a PDF file `report.pdf`, which is your report file written in English

- an optional text file `MEDAL`, in case you want to use the gold medals, with the number of medals listed as a single number in the first line of the file. Use your medals wisely—usage cannot be retracted

Please do not include any other files (e.g. class files) in the GIT repository.