

Homework #2

RELEASE DATE: 03/14/2013

DUE DATE: 03/28/2013, 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

The POOCasino is ready to expand. You, as its core programmer, intends to implement the popular “Deuces Wild” game in the casino. “Deuces Wild” is a single-player video poker game. Please check http://en.wikipedia.org/wiki/Video_poker#Deuces_Wild for some information about the game. The game is played with P-credits—virtual money in the POO BBS System. Upon entering the game, the computer would first show the information of you, the sacred author. Then, the player will be asked to enter her/his name and will be given 1000 P-credits. Each round of the game can be roughly described as follows:

- (1) The computer virtually opens a new deck of 52 cards, and shuffles it.
- (2) The player can choose to bet 1 to 5 P-credits for the round. The P-credits are immediately deducted by the computer.
- (3) The computer distributes the player 5 cards from the shuffled deck.
- (4) The player can choose to keep none, 1, 2, \dots , to 5 cards out of those 5 on hand. Then, other cards are thrown away and replaced with new cards from the shuffled deck.
- (5) The computer then determines the best hand to describe the final 5 cards, where any card numbered “two”, called “deuce”, can substitute for any of the 52 card in order to make a better poker hand, except for the “natural royal flush” hand. The hands, sorted from the “best” to the “worst”, are as follows:
 - natural royal flush: A, K, Q, J, 10 with exactly the same suit, **without using any deuce cards**
 - four deuces: as the name suggests, four deuce cards and a card of another value
 - wild royal flush: A, K, Q, J, 10 with exactly the same suit
 - five of a kind: five matching cards of one value
 - straight flush: five cards of sequential values (a.k.a. ranks) and exactly the same suit
 - four of a kind: four matching cards of one value and an unmatched card of another value
 - full house: three matching cards of one value and two matching cards of another value

- flush: all five cards are of the same suit
- straight: five cards of sequential values
- three of a kind: three matching cards of one value and two unmatched cards of other values
- others

For straight-related hands, A2345 to TJQKA are allowed.

- (6) The computer then pays the user **payoff** P-credits, determined by the P-credits bet by the user and the payoff table below.

hand	1 P-credit	2 P-credits	3 P-credits	4 P-credits	5 P-credits
natural royal flush	300	600	900	1200	4000
four deuces	200	400	600	800	1000
wild royal flush	25	50	75	100	125
five of a kind	15	30	45	60	75
straight flush	9	18	27	36	45
four of a kind	5	10	15	20	25
full house	3	6	9	12	15
flush	2	4	6	8	10
straight	2	4	6	8	10
three of a kind	1	2	3	4	5
others	0	0	0	0	0

The screen output of your program will be something like

```
P00Casino Deuces Wild, written by b86506054 Hsuan-Tien Lin
Please enter your name: CharlieL
Welcome, CharlieL.
Your have 1000 P-credits now.
Please enter your P-credit bet for round 1 (1-5 or 0 for quitting the game): 3
Your cards are (a) SA (b) HA (c) D5 (d) C3 (e) C2
Which cards do you want to keep? acde
Okay. I will discard (b) HA.
Your new cards are SA D4 D5 C3 C2.
You get a straight hand. The payoff is 6.
Your have 1003 P-credits now.
Please enter your P-credit bet for round 2 (1-5 or 0 for quitting the game): 0
Good bye, CharlieL. You played for 1 round and won 3 P-credits.
```

You are asked to use the following classes to implement the game.

- a “Card” class that represents one of the 52 cards in a standard deck of playing cards
- a “Shuffler” class that implements the shuffling of N decks as described above (hint: read or use the `RandomIndex.java` on the course website)
- a “Computer” class that contains the actions of the computer
- a “Player” class that interacts with the actions from the human player
- a “POOCasino” class that demonstrates the game

The structure above is only a suggestion. If you think you have a better design, you can contact the TAs for a quick approval of your own program structure and illustrate them in your report for bonus points.

We expect you to use suitable **class definitions**, **constructors**, **instance variables**, **instance methods**, **static variables/methods** and **encapsulation** by `public/private` to finish the task. All those components are covered by the course materials on or before the 03/18 class.

2 Requirements

- Read the document of the `ntu.csie.oop13spring.Card` class (source code NOT provided), and use it in your program.
- Write the code that implement the classes described above. Note that readability of your source code would be worth 20 points out of 200 this time. That is, your grading TA would give you points based on the following qualitative measure:

20 very readable
16 readable
12 mostly readable, but with some unreadable parts
8 mostly unreadable, but with some readable parts
4 unreadable
0 very unreadable

Also, the TAs will qualitatively justify whether you have appropriately used the components (see the end of Section 1 above) for the task.

- Write a short report with at **most** three A4 pages that contains the following items:
 - (1) your name and school ID
 - (2) how a human player should play with your program
 - (3) how you tested the correctness of your program
 - (4) the output from three rounds of the game from your program
 - (5) any part that you implemented that is worth getting “bonus” points. We allocate at most 25 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. You may get no points for submitting a non-PDF report.

3 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:

- `src/*`, your source code
- `Makefile`, where the TAs can use `make` on CSIE R217 linux machines to compile your code, and then `make run` to test your 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 ZIP.