

FOOP HW4 Report

資工三 B02902031 李彥霆

1. Player Strategy

I implement a simple decision consider only the cards I hold and the dealer's open card, which is almost the same as wiki writes. The only difference is that I don't surrender since I don't know my faced-down card. I also implement a player that we can play with POOCasino as bonus. I believe that there are some better decision based on human intelligence.

2. Design of all the classes

- (1) Arranger : It plays a roll like a really player in the casino. I need this class because the player class we have is only for decision. Without the actual body of a man, it takes a lot more efforts to aid this "brain" to play the game. Such as the decision he made, bet, cards in hand, we have to find someone to remember, so I design the arranger.
- (2) Checker : It contains all method to the operation of class Card and Hand. Since class Card contains only numbers, it takes many lines of codes to print a card and thus make the code more unreadable. I design Checker with all static method to make code more readable.
- (3) Dealer : To make Casino simpler, I make dealer also an independent class. It behaves much like player, but with less function.
- (4) Shuffler : I make the shuffler look much alike the machine with piles of cards. This can make the dividend of job more clearly, and I believe this design can make my code more OO like.
- (5) POOCasino : It is a table of Black Jack in a casino. I make it an instance instead of static method or in main because with instance, we may use multithreading technique to extend the casino. There are always lots of tables with different card games, or even the same game, in the casino. So I think the code should be more flexible. And this design also make my adjusting for bonus part much easier.

3. Result of the duel between me and my classmate(B03902084).

```
=====Round 2 start=====
Ask players to bet...
B02902031 at 1 bets 4.
B02902031 at 2 bets 5.
B03902084 at 3 bets 33.
B03902084 at 4 bets 5.
Assign cards to players...
B02902031 at 1 gets : S4
B02902031 at 2 gets : SQ
B03902084 at 3 gets : C2
B03902084 at 4 gets : CK
Dealer gets : H7
Dealer do not get a Black Jack.
Ask players whether to surrender...
B02902031 at 1 decides not to surrender.
B02902031 at 2 decides not to surrender.
B03902084 at 3 decides not to surrender.
B03902084 at 4 decides not to surrender.
Flip up B02902031 's faced-down card
B02902031 at 1 now has : S4 HA
B02902031 at 1 decides not to double.
B02902031 at 1 decides to hit.
B02902031 at 1 now has : S4 HA D9
B02902031 at 1 decides to hit.
B02902031 at 1 now has : S4 HA D9 D7
B02902031 at 1 decides to stand.
B02902031 now have 21 points
Flip up B02902031 's faced-down card
B02902031 at 2 now has : SQ DA
B02902031 at 2 decides not to double.
B02902031 at 2 decides to stand.
B02902031 now have 21 points
Flip up B03902084 's faced-down card
B03902084 at 3 now has : C2 D2
out of money!!!
can't split
B03902084 at 3 decides not to double.
B03902084 at 3 decides to hit.
B03902084 at 3 now has : C2 D2 H10
B03902084 at 3 decides to hit.
B03902084 at 3 now has : C2 D2 H10 SA
B03902084 at 3 decides to hit.
B03902084 at 3 is busted : C2 D2 H10 SA CQ
B03902084 now have 25 points
Flip up B03902084 's faced-down card
B03902084 at 4 now has : CK D5
B03902084 at 4 decides not to double.
B03902084 at 4 decides to hit.
B03902084 at 4 is busted : CK D5 S9
B03902084 now have 24 points
Flip up dealer's card
Dealer now has : H7 H2
Dealer decides to hit.
Dealer now have : H7 H2 C8
Dealer decides to stand.
Check the result of each player...
Dealer gets 17 points.
B02902031 at 1 wins 4 chips!!
B02902031 at 2 gets Black Jack and wins 7.5 chips!!
B03902084 is busted
B03902084 is busted
Print status of each player.
B02902031 : 94.0
B02902031 : 107.5
null
null
```

Running with other people's code reminds me that everyone may take different ways to achieve the same goal. The design of casino is vary from classmates. With the cooperation, many bugs that don't appear in my own code has pop up one by one. To say a simple case but makes me impressive, my player originally doesn't anticipate that there would be a hit_me request even when it's already busted. It cause not the loss of my player but the shutting down of the game. It comes to the issue that who should handle the condition? Both maybe correct but finally I think it's two side of the request should handle it. It's redundant but when cooperate with others, to make codes run on the best stability, the out-of-spec condition should carefully handled.

4. The consideration of some condition that is not specified at spec.

- (1) The role of insurance : Our casino is quite different from the common ones.
The payment of insurance sometimes is paid at the moment dealer gets Black Jack, while our design is pay the chips all at the end of the stage. Here comes an issue that if a player buys the insurance and the dealer gets Black Jack and somehow he is also busted, then in the case should he get his insurance payment? For the condition, my implement says yes, considering the insurance can be regarded as another bet between dealer and player, and thus I make it independent from the origin game.
- (2) Can a player double after split? After gathering opinions on the internet, I found that most people say no in their rules. So in my implements, no double can be made after split.
- (3) Determining of Black Jack. The most ambiguous point is that after splitting, does each hand can both be Black Jack? The answer I found out is no. Black Jack is the origin two cards we get in a round, so after any operation toward the cards, that's no more Black Jack.

5. Bonus implements : Online mode !!

- (1) Introduce:
It's a simple implement by java socket and multithreading, and the POOCasino structure stays almost the same. I change few function and member to protected, and most of the codes are about the connections.
Black Jack is more interesting with more players, so lets go play it online!
- (2) Usage : after make the origin POOCasino, make bonus can compile the bonus part. make bonusRun than execute the process with initial 100 chips for everyone and set port number 10000. make client can start the client process, the default address 127.0.0.1.
- (3) Class design:
 - I. ServerPlayer : extends player to fit in the origin structure but with some other function to satisfy the need of communication. The functions of player is a call to the client for decision.
 - II. Request : the object of communicate implements serializable. The origin design try to minimize modification of the object, but then I turn it into string with html-like angle brackets, and implements methods to transform the string to the origin objects

- III. ServerListener : a thread to accept new connection.
- IV. BonusPOOCasino : the main structure of the server.
- V. Client : Handle all the condition at client side.
- VI. PlayerHuman : an interface for user to interact with server. It can also play offline with make Human and make runHuman to start the process.
- VII. due to time limit, do_double is now disabled for online mode.

(4) Results :

```

ralph@Ralph: ~/15fall/FOOP/hw4
javac -cp "lib:/bin/" src/PlayerB02902031.java -d lib
javac -cp "lib:/bin/" src/Arranger.java src/Shuffler.java src/POOCa
sino.java src/Checker.java src/Dealer.java -d bin/
ralph@Ralph:~/15fall/FOOP/hw4$ make bonus
javac -cp "lib:/bin/" src/bonus/*.java -d bin/
ralph@Ralph:~/15fall/FOOP/hw4$ java -cp "lib:bin" BonusPOOCasino 10
0 10000

This is the Bonus POOCasino written by Ralph Lee, ID B02902031.
Ask players to bet...
Player at 1 bets 10.
Assign cards to players...
Player at 1 gets : D7
Dealer gets : D9
Dealer do not get a Black Jack.
Ask players whether to surrender...
Player at 1 decides not to surrender.
Flip up Player 's faced-down card
Player at 1 now has : D7 C2
Player at 1 decides not to double.
Player at 1 decides to hit.
Player at 1 now has : D7 C2 S5
Player at 1 decides to stand.
Player now have 14 points
Flip up dealer's card
Dealer now has : D9 C8
Dealer decides to stand.
Check the result of each player...
Dealer gets 17 points.
Player at 1 lose.
Print status of each player.
Player : 90.0
Ask players to bet...
Player at 1 bets 5.

```

```

ralph@Ralph: ~/15fall/FOOP/hw4
Dealer : D9
Your card is D7 C2 S5 . And Dealer has D9.
Do you want to hit? (y/n) n
You get : D7 C2 S5
Dealer get : D9 C8
You lose.
Result :
Hand1 : D7 C2 S5
Dealer : D9 C8
=====New Game Start=====
Last table :
Hand1 : D7 C2 S5
Dealer : D9 C8
2 players in this turn. You're position is 1
You have 90.0 chips now.
Make bet : 5

```

```

ralph@Ralph: ~/15fall/FOOP/hw4
ralph@Ralph:~/15fall/FOOP/hw4$ java -cp "lib:bin" Client 127.0.0.1
10000
Enter your name : Linda
=====New Game Start=====
Last table :
Hand1 : D7 C2 S5
Dealer : D9 C8
2 players in this turn. You're position is 2
You have 100.0 chips now.
Make bet : 

```