**FOOP HW3- Old-Maid Re-implementation and 2 Variants**

B01902039 施楷文

1. Introduction

In this homework, I modified the program in hw1, making it more object oriented. And there are two variants I made. **The first one** is the basic variant that is on the wiki page, in the game, there are no jokers and there is randomly one card been removed from desk. **The second one** is, there is randomly one card value (2, 3, 4… J, Q, K, A, Joker) that these cards can only be dropped when they matched the value and color (Club = Spade, Heart = Diamond). The other rules remain the same.

2. My design of class structures
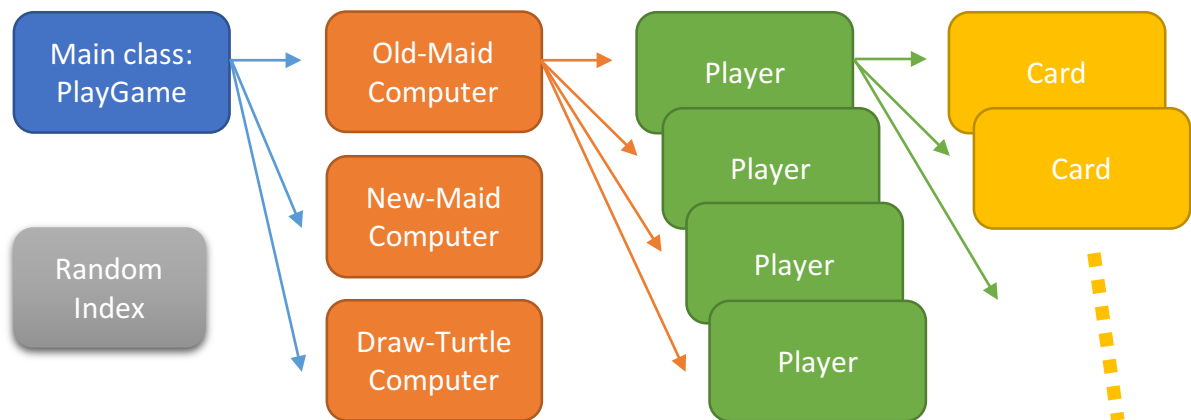
Main class- PlayGame: Call the computer class including the original version and two variants to start and host the game.

```
3  public class PlayGame{
4      public static OldMaidComputer gameHost;
5      public static void main(String[] argv){
6          startGame();
7          gameHost.dealCards();
8          gameHost.dropCards();
9          while(gameHost.isGameOver()!=1){
10             gameHost.drawCards();
11         }
12     }
13     private static void startGame(){…
28     }
29 }
```

Computer class (including the two variants): The computer class have to maintain the game status & the players' status, start/host/stop the game, and the most important job is to communicate with the player.

```java
147   class OldMaidComputer{
148       ArrayList<Player> playerPool;
149       Player player0,player1,player2,player3;
150       int whosTurn;
151
152       OldMaidComputer(){...
163       }
164
165       void dealCards(){...
183       }
184
185       // all player drop cards
186       void dropCards(){...
192       }

194       // one player drops cards
195       private void dropCards(Player p){...
208       }
209
210       void drawCards(){...
222       }
223
224       int isGameOver(){...
270       }
271   }
```

Player class: The class contains the player status, what cards does he have, and also some methods manipulating cards in hand.

```java
273 ▼ class Player{
274       int playerNumber;
275       List<Card> cardsInHand;
276
277       // Initiate the player status
278 ►     Player(int num){...
281       }
282
283 ►     void getCard(int value){...
286       }
287
288 ►     void discardCard(int index){...
291       }

292
293       // Sort arraylist<Card>
294 ►     void sortHand(){...
300       }
301
302 ►     void ShowCards(){...
309       }
310
311 ►     boolean isWin(){...
315       }
316   }
```

Card class: The card class contains the status of one card, including card value in several forms (integer: -2~51, 0 & 2~14, string such as "C10" "SA")

```java
318   class Card{
319       int valueOfCard;
320       int intValue; // 0 for joker, 2~14 for others
321       int color; // 0 for black, 1 for red
322       String card = "";
323
324       // Ininiate the card status
325       Card(int value){...
328       }
329       // Change card raw value into suit & value in string
330       private String ValueToSuitValue(){...
362       }
363   }
```

Shuffler class: I use the class provide by TAs in hw2.

3. How to play with the two variants?

Since there are no big differences, the way to play with my program is quite similar.

a. Open command line.

b. **$make all** in the game directory.

c. **$make run** in the game directory.

d. Type in which game mode you want to play, the rules of the two variant games are quite trivial.

4. How I tested my program?

Since my hw1 program was totally tested, I only modified the structure of the program to make it more object oriented. In this homework, I use the test program (which is provided by my classmate Yeng-Chieh Sung) which is the one I used in hw1 to make sure there are no mistakes in game. And then, to make sure the variants work, I manually manipulated the game to make sure if it follows the rules. I simply add some codes to receive input in order to cut every round separately that I can read the output more clearly.

5. The sample output of each variant

Variant 1:

```
Kevins-MBP:hw3 kevinshih$ make run
java PlayGame
Which game do you want to play (1) Old-Maid (2) New-Maid (3) Draw Turtle: 2
Game start, the super card is 7.
Deal cards
Player0: B0 D3 D4 C5 H6 H7 C10 D10 H10 S10 DJ DQ HQ DK
Player1: H3 S3 C4 D5 H5 S6 C7 D8 D9 S9 SQ HK SK HA
Player2: D2 S2 S5 C6 D6 D7 H9 HJ SJ CQ CK CA SA
Player3: R0 C2 H2 C3 H4 S4 S7 C8 H8 S8 C9 CJ DA
```

```
Player1: C4 S6 C7 D8 CQ
Player1 draws a card from Player2 D7
Player1: C4 S6 C7 D7 D8 CQ
Player2: S5 H9 CJ CK
```

Variant 2:

```
Kevins-MBP:hw3 kevinshih$ make run
java PlayGame
Which game do you want to play (1) Old-Maid (2) New-Maid (3) Draw Turtle: 3
Game start, the turtle card card is HJ. (players don't know)
Deal cards
Player0: S2 D3 S5 C6 D6 D7 C9 H9 C10 CQ SQ HK SK
Player1: H3 H4 H5 H6 C7 H7 S7 H8 S9 D10 S10 CJ DA
Player2: D2 C3 S3 D4 S4 D5 C8 S8 D9 DJ DQ CK CA
Player3: C2 H2 C4 C5 S6 D8 H10 SJ HQ DK HA SA
```

```
Player2 draws a card from Player3 S2
Player2:
Player3: SJ
Player2 wins
Bonus game over
```