FOOP—HW2 Report
Name: 江東峻
Student ID: B01902032

1. How to play

To start the program, use “make run” or something like “java POOCasino 0”, argument 0 means starting with the normal mode.

(1) Enter the player name

First, the program will show the information of the author, and request the player’s name.

When “Welcome, [player’s name]” shows up, the game starts. The computer will tell you the P-dollars you have. Initially, every player has 1000 P-dollars.

(2) Each round starts

(a) Enter your bet for this round (bet = 0 ~ 5), or enter 0 for quitting the game.

(b) After the player enter the legal bet, the bet will be immediately deducted from the P-dollars you have.

(c) The computer distributes the player 5 cards from the shuffled deck. And each card will be given an ID.

(d) Enter the IDs you want to keep. The computer will keep those cards of IDs and drop the remaining cards. (the cards discarded will be shown.) And the computer distributers the player cards until the player have 5 cards.

(e) The computer shows your current hand, finds the best hand type, and pays the payoff according to your bet and hand type.

(f) The computer tells you the P-dollars you have.

(3) If you quit the game, the computer will say Goodbye to you, and tell you the rounds you have played for and the remaining P-dollars.

2. How to test the correctness

It is hard to test the correctness of this program because its shuffler randomly distributes cards. So, I write a test mode to manually enter the cards:

(1) To start the program with the test mode, use “make test” or something like “java POOCasino 1”.

(2) For shuffled deck, I dump all cards in shuffled deck and check the number and whether there is a duplicate card or not.

(3) For distributed cards, I write some function to manually enter the hand cards. By this way, I can create lucky conditions like “royal flush” and easily check the correctness of the following procedures like “find best hand” or “find the payoff”.

(4) To check the connection of these functions, I can dump the deck but not manually enter the hand cards. This method make sure my “distribute cards” function is not broken.

3. The output from three rounds of the game

The player name: Gdog
Output for 3 rounds:

```
POOCasino Jacks or better, written by b01902032 Chiang Tung-Chun (Gdog).
Please enter your name: Gdog
Welcome, Gdog.
You have 1000 P-dollars now.
Please enter your P-dollar bet for round 1 (1-5 or 0 for quitting the game): 5
Your cards are (0) D10 (1) D8 (2) C9 (3) D4 (4) C5
Which cards do you want to keep? (none or index) 013
Okay. I will discard (2) C9 (4) C5
Your cards are (0) D10 (1) D8 (2) D5 (3) D4 (4) CJ
You get a [others] hand. The payoff is 0.
You have 995 P-dollars now.
Please enter your P-dollar bet for round 2 (1-5 or 0 for quitting the game): 4
Your cards are (0) DK (1) SJ (2) SK (3) C6 (4) CQ
Which cards do you want to keep? (none or index) 02
Okay. I will discard (1) SJ (3) C6 (4) CQ
Your cards are (0) DK (1) DQ (2) SK (3) CJ (4) C7
You get a [Jacks or better] hand. The payoff is 4.
You have 995 P-dollars now.
Please enter your P-dollar bet for round 3 (1-5 or 0 for quitting the game): 5
Your cards are (0) C9 (1) S4 (2) D9 (3) S8 (4) C3
Which cards do you want to keep? (none or index) none
Okay. I will discard (0) C9 (1) S4 (2) D9 (3) S8 (4) C3
Your cards are (0) DQ (1) C4 (2) C5 (3) SQ (4) CJ
You get a [Jacks or better] hand. The payoff is 5.
You have 995 P-dollars now.
Please enter your P-dollar bet for round 4 (1-5 or 0 for quitting the game): 0
Good bye, Gdog. You played for 3 rounds and have 995 P-dollars now.
```

4. To get “bonus” points

(1) I try to use some “private” variables in some class. For example, the shuffled deck is a private variables because we can not access the deck or you can see the next card and make a better decision. Reasons for other examples are commented in the code. The main idea is to model the real computer and player. For example, you can not know a player’s money without asking, so the P-dollars a player has is a private variable.

(2) I use ArrayList rather than array to store Card. To distribute cards, the solution using array needs two variables (array and a integer for stack top), but the solution using ArrayList can use remove method (remove(0) to pop). And we can use Collection.shuffle method to replace a Shuffler class. It is more clear and short for code.

(3) The computer will ckeck if the bet you enter is legal (0-5), it will ask you to enter again if you type a illegal bet. The Player class check if your P-dollars are enough for the bet you enter. If you don’t have enough P-dollars, the computer will say Goodbye to you. I implement this checking in Player class because in real world, the computer only knows the tokens you enter, and this checking is from a human’s knowledge.