

CECS 277 HOMEWORK ARRAYLISTS

OBJECTIVE: Get some experience using ArrayLists as well as OO design.

INTRODUCTION: Please remember the coding standards [here](#).

You will build an application that simulates the game of War between two players. As you will see, this game has no decisions that the players make. Once the two players receive their cards, the eventual outcome of the game is completely determined. You need to build an application that randomly deals out the 52-card deck to two players, and then simulates those two players playing War with each other. The rules of the game are:

- War is a card game for two players.
- A standard deck of 52 cards is dealt so that both players have 26 cards.
- During each round of play (or "battle"), both players play a card from the top of their hand face up.
- The player who plays the card of the higher rank wins both cards and places them at the bottom of his stack of cards.
- If both cards played are of the same rank, then both players play three additional cards face down and then one more card face up (this is called a "war").
 - 2 of any suit is less than 3 of any suit, and so forth up through 10.
 - Jack beats a 10
 - Queen beats a Jack
 - King beats a Queen
 - Ace beats a King
- The player who wins the war by playing the higher card wins all ten cards.
- If the ranks are still the same, additional wars are played until one player wins the turn.
- If either player runs out of cards to play, he loses the game.
- You will use only ArrayLists to store the cards in this program. This means cards cannot be stored in arrays.

Sample output might look like this:

```
player 1 plays Card is 9 of Clubs
player 2 plays Card is Ace of Diamonds
player 2 wins the round
player 1 plays Card is 10 of Diamonds
player 2 plays Card is Ace of Hearts
player 2 wins the round
player 1 plays Card is 7 of Clubs
player 2 plays Card is 6 of Hearts
player 1 wins the round
player 1 plays Card is 9 of Hearts
player 2 plays Card is 9 of Clubs
war
war card for player1 Card is xx
war card for player2 Card is xx
```

CECS 277 HOMEWORK ARRAYLISTS

```
war card for player1 Card is xx
war card for player2 Card is xx
war card for player1 Card is xx
war card for player2 Card is xx
war card for player1 Card is 7 of Clubs
war card for player2 Card is 10 of Diamonds
player 2 wins the war round
player 1 plays Card is 6 of Hearts
player 2 plays Card is Ace of Hearts
player 2 wins the round
game over
player 2 wins the game
```

- You may find methods in the Collections class that will help with some of the functionality needed in this project.
- Make sure you document the programs as specified in the grading guidelines.
- You will hand in a printout of the program(s) along with a demo in the lab.

The question was asked if both players have the same number of cards in their hand, and neither one is able to complete the three cards down and one card up for the next tie-breaking war, what happens. To get there, you would have to have a very long string of ties that occurred in a row, since both players would have half of the deck if they had the same number of cards. So, if you want to explicitly treat such a case, I would say that the game ends in a draw. On the other hand, if one player runs out of cards before the other one does while trying to play three cards down and one up, then the one who has the most cards wins.

Another question was whether to put a cap on the number of plays in a single game. If you have played war before, you have noticed that it's entirely possible for the game to go on for an extended period of time, with the odds seemingly favoring one player, then the other. I suggest that you put a cap on the number of plays that can be executed before the game is suspended in a draw. Do not make that a literal; make it a parameter that is passed into the program at run time, or even read from a parameter file if you really want to get frisky.

PROCEDURE:

1. Think of a Card as an object with properties (rank and suit).
2. Think of a Deck of Cards as a collection of Cards.
3. Think of the remaining objects: Hand, Discard Pile, etc. as a collection of some objects.
4. You may find methods in the Collections class that will help you with some of the functionality needed in this project.
5. Make sure that you document the programs as specified in the [coding standards](#).

WHAT TO TURN IN:

CECS 277 HOMEWORK ARRAYLISTS

- All of the code that you write for this assignment.
- Your sample output.
- A demonstration during lab.