Griffin's OH 4/2/22

COMS 1004 Introduction to Computer Science and Programming in Java

Quick Announcements

Announcements for the week of April 2nd 2022

- You have made it to the last month of classes, classes end May 2nd 2022
- Quiz 5 will be on April 14th so study now if you deem it necessary
- The registrar has projected dates for the final, they are:

Tuesday May 10th from 7:10pm to 10:00pm for section 1 and **Thursday May 12th from 4:10pm to 7:00pm**

- Currently the exam is planned to be in-person and written, but is always subject to change and at professor's discretion

Topics for the Week

1. Programming Project 4 Overview
2. Card.java Overview
3. Deck.java Overview
4. Player.java Overview

1

Programming Project 4 Overview

Programming Project 4 Overview

Here are the rules for implementation:

1. Seven cards are dealt to each player
2. The remaining cards are placed in a pile face down. This pile is called the “stock pile”
3. The top card on the stock pile is removed and turned face up. It is now the first card in a new pile called the "discard pile". (Notice the stock pile cards are face down and the discard pile is face up.)
4. Each turn a player discards a card from their hand into the discard pile by matching either the suit or the rank of the top card of the discard pile. If the player has no cards in their hand that match the suit or the rank, they must draw from the stock pile until they get a playable card or they may play an eight. The player’s discarded card is now the top card of the discard pile.
5. Eights are wild. That means an eight may be played at any time and the player discarding the eight then chooses the suit they wish the next player to have to match.
6. If a player discards the final card in their hand, they win.
7. If the stock pile is exhausted, the game ends and the player with the fewest cards left in their hand is declared the winner.

Common FAQs

- *Can we assume that the user input is in the correct format?*
  You can assume the user is well behaved. That is, you can assume that user will not input numbers or strings that are out of range and will generally follow formatting instructions when providing input.
- *Can we assume the user does not try to cheat by playing cards that don't match the up card?*
  You may make this assumption and still earn full marks for "running properly" and "design". To earn full marks in "extra" the user must not be able to play cards that the game rules prohibit.
- *Can we assume that the user does not try to cheat by playing cards that are not in their hand*
  You may make this assumption and still earn full marks for "running properly" and "design". To earn full marks in the "extra" category the user must not be able to play cards that are not in their hand.

2

Card.java Overview

Card.java Overview

This is what your class needs to do/have:

1. Constructor with proper parameters (you'll need to be taking into account the suit and the rank)
2. You'll need accessors for the suit and rank
3. Proper toString() method(s) for the class

Think about what instance variables you'll need for this data and what the types will need to be.

2

Deck.java Overview

Deck.java Overview

This is what your class needs to do/have:

1. A proper Constructor (think of the instance variables you'll need to have)
2. A method to deal which returns a Card object
3. A method to see if you can deal
4. A method to shuffle the Cards in the deck (there are multiple ways to do this)
5. A proper toString method

3

Player.java Overview

Player.java Overview

This is what your class needs to do/have:

1. Proper constructor be sure to instantiate the proper instance variables
2. Method to add cards to player's hand
3. Proper accessor methods as necessary to
4. A Method that plays a player's card
5. A Method that handles logic for player turn
6. Proper toString method

Link to Video Drive and Other Resources

Link to Video Drive:
Video Drive

Link to My Office Hour Materials:
Useful Files