

CS111 - Fall 2017 >   Announcements

Announcements

VIEW

CS111 Project Description (Milestone 2)

Saved By

Andrew Tjang

Modified Date

Nov 30, 2017 11:30 pm

Beginning Date

Dec 1, 2017 5:00 pm

Groups

site

Message

In this milestone, you will create a Player class to model the behavior of a poker player. Each poker player must keep track of his/her hand, fund balance, and current bet. Implement the following methods in Player.java. We have provided you with a Player.java file to fill in.

The original Hand.java you were given had a bug in the removeCard() method. Please replace your removeCard(int position) in Hand.java method with the one that follows:

```
public void removeCard(int position) {
    if (position < 0 || position >= hand.length)
        throw new IllegalArgumentException("Position does not exist in hand: " + position);
    hand[position] = null;
}
```

The methods for Player.java are reproduced below for your convenience. Also for your convenience, a Driver will be released on Monday to help you test your code. (write your own program to test your code - it's good practice!)

The autograder will open on Wed of next week 12/6, and this milestone is due on Fri 12/8.

```
//initialize your fields in the constructor
public Player(double balance){

}

public void deal(Card c){
```

```
}

//Returns an array of Cards that the Player wishes to discard.
//The game engine will call deal() on this player for each card
//that exists in the return value. MS2 Instructions: Print the hand to
//the terminal using System.out.println and ask the user which cards
//they would like to discard. The user will first the number of cards they
//wish to discard, followed by the indices, one at a time, of
//the card(s) they would like to discard,
//Return an array with the appropriate Card objects
//that have been discarded, and remove the Card objects from this
//player's hand. Use IO.readInt() for all inputs. In cases of error
//re-ask the user for input.
//
// Example call to discard():
//
// This is your hand:
// 0: Ace of Hearts
// 1: 2 of Diamonds
// 2: 5 of Hearts
// 3: Jack of Spades
// 4: Ace of Clubs
// How many cards would you like to discard?
// 2
// 1
// 2
//
// The resultant array will contain the 2 of Diamonds and the 5 of hearts in that order
// This player's hand will now only have 3 cards
public Card[] discard(){

}

//Returns the amount that this player would like to wager, returns
//-1.0 to fold hand. Any non zero wager should immediately be deducted
//from this player's balance. This player's balance can never be below
// 0 at any time. This player's wager must be >= to the parameter min
// MS2 Instructions: Show the user the minimum bet via the terminal
//(System.out.println), and ask the user for their wager. Use
//IO.readDouble() for input. In cases of error re-ask the user for
//input.
//
// Example call to wager()
//
// How much do you want to wager?
// 200
```

```
//  
// This will result in this player's balance reduced by 200  
  
public double wager(double min){  
  
}  
  
//Returns this player's hand  
public Hand showHand(){  
  
}  
  
//Returns this player's current balance  
public double getBalance(){  
  
}  
  
//Increase player's balance by the amount specified in the parameter,  
//then reset player's hand in preparation for next round. Amount will  
//be 0 if player has lost hand  
public void winnings(double amount){  
  
}
```

Attachments

-  [Player.java](#) (2 KB)

[< Previous](#)[Return to List](#)[Next >](#)

-
- [Teaching and Learning with Technology](#)
 - Contact Sakai Helpdesk Support: sakai@rutgers.edu
 - 848.445.8721
 - [Report an Accessibility Barrier](#)
 - [Rutgers University](#)
 - Copyright 2003-2019 The Apereo Foundation. All rights reserved. Portions of Sakai are copyrighted by other parties as described in the Acknowledgments screen.

CS111 - Fall 2017 >   Announcements

Announcements

VIEW

CS111 Project Description (Milestone 3) (Corrected)

Saved By

Andrew Tjang

Modified Date

Dec 9, 2017 1:05 am

Groups

site

Message

Milestone 3

Now, you will create an Artificial Intelligence(AI) poker player. Using your best poker strategy, implement the `wager()` and `discard()` methods in MS2. At the very least (for full credit), your implementation **MUST** return valid values for these two methods.

Put into other words, in this milestone, remove all prompts and IO library calls from the above mentioned methods. Then, using some strategy/algorithm, return a valid wager from `wager()` and a valid set of cards to discard in `discard()`. You can use the same driver as given in MS2.

Submit your AI poker player as ~~Poker.java~~ **Player.java** to the autolab grader. The grader will be released early next week and due on Wednesday 12/13 5pm. We will continue accepting submissions without penalty until 12/14 11:59pm.

AI Player Contest (optional)

A second, non scored autograder will be released as well. This autograder will rank AI players submitted by all students who wish to participate. I will give out a prize to the student who codes the AI player who has the highest balance at the end of 100 games. The prize is guaranteed to be crappy. You should not enter this contest just for the prize. You should enter the contest to challenge yourself.

Here are the rules of the poker game that your AI player will be playing.

1. Each player will be given an initial balance of \$10,000
2. Each player will be dealt a hand of 5 cards
3. Initial wager will start with the first player (first player will be alternated in each round)
 - Our AI player will always initially bet some amount of money > 0
 - Our AI player will randomly match a bet of yours
 - Our AI player will have some wager threshold over which it will fold, dependent on its hand.

4. Wagering will continue back and forth between the two players until both players “stand” (i.e. wager 0)
5. Each player will have an opportunity to discard 0 – 5 cards in their hand
6. Each player will be dealt new cards, replacing the cards they discarded
7. A second round of wagering will occur identical to (3) – (4)
8. The game will determine the winner based on the hands and distribute the winnings to each player

Prior to any game play, the autograder will ensure that you are not “cheating”. (i.e. it will ensure that your discard, winning, and wager functions adhere to the original specifications. Any submissions that do not pass these initial checks will be disqualified from placing.

[< Previous](#)[Return to List](#)[Next >](#)

-
- [Teaching and Learning with Technology](#).
 - Contact Sakai Helpdesk Support: sakai@rutgers.edu
 - 848.445.8721
 - [Report an Accessibility Barrier](#)
 - [Rutgers University](#).
 - Copyright 2003-2019 The Apereo Foundation. All rights reserved. Portions of Sakai are copyrighted by other parties as described in the Acknowledgments screen.