

Final Project Proposal
Texas-Hold'em

Our project is to mimic a Texas Hold 'em Poker game between 2-4 players.
Here's a rough outline of our instance variables / methods:

Table Class

Variables

- private final String deck - string array of all 52 cards.
- tableAmount; -- the bet that people have to reach to stay in the pool
- poolAmount; - total amount of money

Methods

- dealerAction() - Either get one or three cards or compares best hand. Calls the following methods:
 - getCard() - get a card from the deck, making sure there aren't any duplicates
 - betterHand() - compares each hand, takes the better hand and gives award money to better hand
 - reset() - reset tableAmount/poolAmount
 - blinds(); - rotate blinds for players

//Character Class

Player Class

Variables

- private double betTotal - how much the player currently bet.
- double balance - the amount of money the player has.

Methods

- blindAmount() - sets betTotal beginning of hand.
- private String[] hand - the player's hand.
- public bet(double betAmount) - reduce balance by amount, increases poolAmount in class Table by amount.
- public getBalance() - accessor of balance.
- public getBetTotal() - accessor of betTotal.
- getHand() - accessor of hand.
- compareTo(player p) - compares hand of current player to that of other player p.
- abstract action(double currentNeeded) - takes current needed to stay in pool. Calls the following methods:
 - fold() - sets betTotal to 0 and removes player from round.
 - call() - sets betTotal to tableAmount in class Table.
 - Check would be calling, it would just maintain all values
 - raise(double raiseAmount) - raises tableAmount in class Table by difference between tableAmount and betTotal.

//Player Class

- //Extends Character class.
- Imports cs1/Keyboard.java
- printHand() - prints the current hand of player.

Cpu Class

- Extends Character class.

Minimum Viable Product

5 random cards returned, the user would just have check. Our compareTo/showdown will focus on pairs at first and the program will return the highest pair if you have a pair.

Our product will be using arrays and the Comparable interface to create a poker game checkup.

User Experience

We want to make our user experience similar to what we did for Character.java/Ye Old Role Playing Game--we take inputs from the keyboard for what they want to do, such as setting the number of players, or declaring what they want to do for their action. For instance, on execution, they will be prompted to enter a number from 2-4 to set the number of players, and can then use numbers 1-4 to declare their move.

Stretch

We plan to spend a lot of time debugging. We also plan to work a lot on how we're going to implement AI along with the Cpu, and generally getting the runtime to go smoothly.

We think the above implementation would be our core plan for our project, but that being said, if we have extra time, we would focus on building our Cpu class (different difficulties of the Cpu, having it calculate % chance it has best hand, taking in account of how players bet per hand and in total, time other players take a turn). We'd also want to implement a game where there are multiple bots playing against each other and the player. We think this is one of the most exciting parts of our project. The AI could keep information on players so it would have better odds against you as you play the game.

A less exciting but more easily implementable add-on would be to add options for the player in terms of pot size ie "½ pot" and "pot" for calling, in addition to call amount.