

CSCI 135 Final
Programming portion part 2

Submit your program files to Moodle

Your program will be graded on correctness and to a lesser degree on clarity (including comments) and efficiency. You will lose a substantial number of points if your code does not compile or crashes on typical inputs.

2 card players have been playing a game a long time. A deck of cards contains cards with for suits H, S, D, and C. Each suit contains 13 cards 01 through 13.

```
"H01","H02","H03","H04","H05","H06","H07","H08","H09","H10","H11","H12","H13","S01","S02","S03",  
"S04","S05","S06","S07","S08","S09","S10","S11","S12","S13","C01","C02","C03","C04","C05","C06","C0  
7","C08","C09","C10","C11","C12","C13","D01","D02","D03","D04","D05","D06","D07","D08","D09","D1  
0","D11","D12","D13"
```

Each time they play they bring up their old scores and start with a new deck in a random order (shuffled). Each player draws a card. Whoever has the higher number wins 3 points or if they tie they each get one point. Then they discard their card to a separate pile.

Your goal is to automate this game for them.

Remember to help prevent bugs and to make them easier to fix code small portions at a time and test.

40 pts

Submit the following 2 programs.

Deck.py

CardClient.py

Deck

Have at least 2 attributes using the following names to hold the deck of cards and the discarded Cards

List cardDeck

List discardedCards

Must have the following constructor and methods

Constructor

Fills cardDeck with all 52 cards in a random order

discardedCards starts empty

getDeck()

returns cardDeck

getDiscard()

returns discardedCards

drawCard()

removes a card from the cardDeck and returns it

discardCard(String)

adds a card represented by the string argument to the discardedCards

nothing happens to the cardDeck

example if the parameter passed in is "HO5", "H05" discardedCards list

shuffleDeck()

removes all cards from discardedCards

adds them to cardDeck

shuffles cardDeck

CardClient

Takes one command line argument of a file name. Look at scores.txt for an example.

Must have one function to read the two players scores from a file with the name provided at the command line. Use a try block to catch any bad file reading. Handle the error however you wish or exit the program once caught.

Read 2 players scores from a file each on their own line and print them.

Play the game by simulating the players taking 26 turns (until the deck is empty) Add the points to their scores for whoever wins or ties. Discard the cards after they are drawn. And finally shuffle the discarded cards back into the deck once done.

Repeat this 4 times.

Print the final scores.

Scores.txt

500

800

Example:

```
C:\Users\brent\Desktop>python CardClient.py scores.txt
500
800
New Scores:
620
923
```

CSCI 135 Final
Programming portion part 2