Lab 10 - 4/3/2019

1. Start from the provided code:

Card.py : A complete version of the Card, Deck and Hand classes in this chapter.

PokerHand.py : An incomplete implementation of a class that represents a poker hand, and

some code that tests it.

2. If you run PokerHand.py, it deals seven 7-card poker hands and checks to see if any of them

contains a flush. **Read this code carefully before you go on.**

3. Add methods to PokerHand.py named has\_pair, has\_twopair, etc. that return True or

False according to whether or not the hand meets the relevant criteria. Your code should

work correctly for “hands” that contain any number of cards (although 5 and 7 are the most

common sizes).

4. PokerHand contains an incomplete method named classify that figures out the highest-value classification for a hand and sets the label attribute accordingly. It currently only checks for a flush, but it is here where you should add additional checks for all the hand rankings. A hand may have more than one possible label, but you should only assign it the highest possible ranking. For example, a 7-card hand might contain a flush and a pair; it should be labeled “flush”.

5. When you are convinced that your classification methods are working, the next step is to estimate the probabilities of the various hands. Write a function in PokerHand.py that shuffles

a deck of cards, divides it into hands, classifies the hands, and counts the number of times

various classifications appear.

6. Print a table of the classifications and their probabilities. Run your program with larger and

larger numbers of hands until the output values converge to a reasonable degree of accuracy. Compare your results to the values at

[https://en.wikipedia.org/wiki/Poker\_probability#Frequency\_of\_7-card\_poker\_hands](https://en.wikipedia.org/wiki/Poker_probability" \l "Frequency_of_7-card_poker_hands)