BIS 420 PROGRAMMING FOR DATA SCIENCE
PRAJAKTA POHARE
CHAPTER 18 EXERCISE 18.6
ILLINOIS STATE UNIVERSITY

The following are the possible hands in poker, in increasing order of value (and

decreasing order of probability):

pair: two cards with the same rank

two pair: two pairs of cards with the same rank

three of a kind: three cards with the same rank

straight: five cards with ranks in sequence (aces can be high or low, so Ace-2-3-4-5 is a straight

and so is 10-Jack-Queen-King-Ace, but Queen-King-Ace-2-3 is not.)

flush: five cards with the same suit

full house: three cards with one rank, two cards with another

four of a kind: four cards with the same rank

straight flush: five cards in sequence (as defined above) and with the same suit

The goal of these exercises is to estimate the probability of drawing these various hands.

1. Download the following files from http: // thinkpython. com/ 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. Write a method named classify that figures out the highest-value classification for a hand and sets the label attribute accordingly. For example, a 7-card hand might contain a flush and a pair; it should be labeled "flush".178 Chapter 18. Inheritance

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 http: // en. wikipedia. org/ wiki/ Hand_ rankings .

```
/usr/local/bin/python3 "/Users/prajaktapohare/Library/CloudStorage/OneDrive-ILStateUniversity/BIS420/Week 18/BIS420_PrajaktaPohare_Ch18_18.6.py"
● → ~ /usr/local/bin/python3 "/Users/prajaktapohare/Library/CloudStorage/OneDrive-ILStateUniversity/BIS420/Week 18/BIS420_PrajaktaPohare_Ch18_18.6.py"
0
1000
2000
3000
4000
5000
6000
7000
8000
9000
70000.0 hands dealt:
straightflush happens one time in 2692.31
fourkind happens one time in 619.47
fullhouse happens one time in 37.31
flush happens one time in 32.54
straight happens one time in 20.12
threekind happens one time in 12.89
twopair happens one time in 3.77
pair happens one time in 1.27
highcard happens one time in 1.00
○ → ~
```