HW-3
CS-595

NAME - SHREYAS SRINIVASA
BLAZER ID - SSRINIVA

1. Ans: $\#(A \cup B) = \#A + \#B - \#(A \cap B)$

The principle of Inclusion - Exclusion is an equation that demonstrates this point.

Consider two sets A and B that have cardinalities of $\#(A)$ and $\#(B)$ respectively. $\#(A \cap B)$ will be used to represent their intersection.

~~∴ According to the principle of Inclusion~~

$\#(A \cup B)$ is equal to the total number of distinct items in both sets and represents the number of elements in the union of A & B.

∴ According to the principle of Inclusion - Exclusion:

$$\#(A \cup B) = \#(A) + \#(B) - \#(A \cap B)$$

2. Ans:- A full house consist of two cards of one rank & three cards of another rank.

To find the number of possible full houses, we can first choose the rank of the three cards, which can be done in 13 ways.

Then we choose the suit for those three cards, which can be done in 4 choices for each card, for a total of $4 \times 4 \times 4 = 64$ ways.

Next, we choose the rank of the pair, which can be done in 12 ways.

The number of ways to choose 5 cards from a deck of 52 cards is :- $^{52}C_5$

$= 2,598,960$

∴ The number of 5 card hands with a full house is $13 \times 12 \times 4 \times 6 = 3744$

Pseudo code :-

```
# Function to calculate the number of full
  house hands
from math import factorial
def fullhands():
    rank = 13
    two_kind = 12
    three_kind = 4
    two_in_four = 6
    full_house = rank * two_kind * three_
                 kind
                 * two_in_four
    return full_house
# To calculate 52C5 combination
def choose(n, j):
    Combi = factorial(n) //(factorial(j)*
            factorial(n-j))
    return Combi
```

```
# Main function
def main():
    full = fullhands()
    total = choose(52, 5)
    prob = full / total
    print("Probability that a randomly drawn 5 card
          hand is a full house: ", prob)

# Calling main
main()
```

3. Ans: No, it is impossible to divide 950 balls into 100 bins such that none of them contain 10 or more balls.

Proof:- The Pigeonhole principle states that at least one bin must contain more than one ball, if there are m bins and n balls, where $n > m$.

$n = 950$ balls and $m = 100$ bins in this scenario.

At least one bin must contain more than 9 balls since $n > m$.

And since no bin can hold 10 or more balls, this breaks the rule.

∴ It is impossible to divide 950 balls into 100 bins, where no bin has 10 or more balls.

4. Ans:- The events that lead to the correct path are:-

(i) Both dogs agree on the correct path and consequently, the hunter picks the correct path.
Probability $\frac{1}{2}$ by independence.

(ii) The dogs disagree, dog 'A' chooses the correct path, and the hunter follows dog 'A'.
So, probability $\frac{1}{2} P[P-1]$

(iii) The dogs disagree, dog 'B' chooses the correct path and the hunter follows dog B.
So, probability $\frac{1}{2} p(p-1)$.

We know that the above events are disjoint, so the probability that the hunter chooses the correct path is :-

$$p^2 + \frac{1}{2} p(p-1) + \frac{1}{2} p(p-1) = p$$

On the other hand, if the hunter lets one dog choose the path, this dog will also choose the correct path with probability p.

∴ The two strategies are equally effective.

HW3_Q2.py

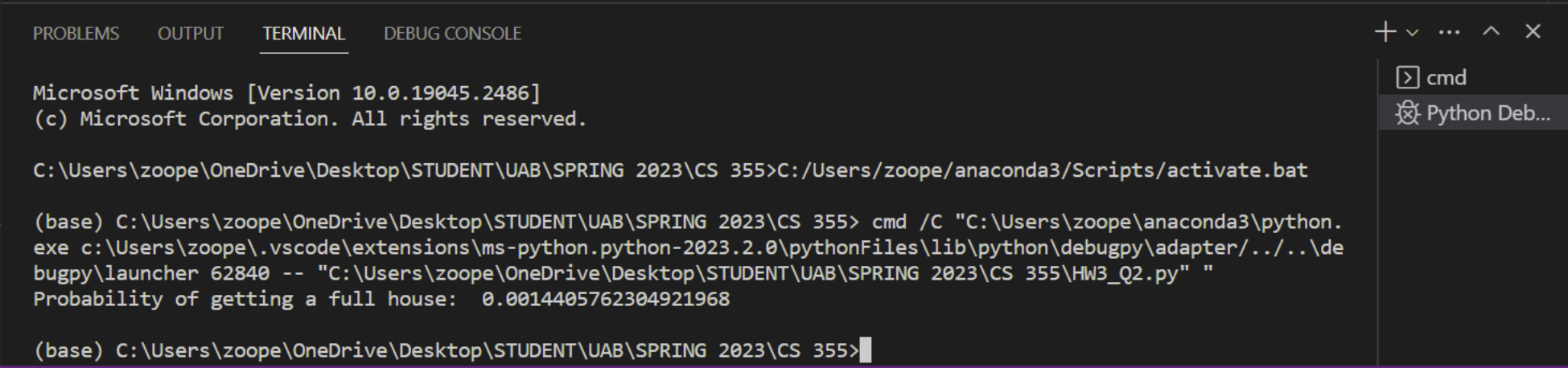
C: > Users > zoope > OneDrive > Desktop > STUDENT > UAB > SPRING 2023 > CS 355 > HW3_Q2.py > main

```python
from math import factorial
# Function to calculate the number of full house hands
def fullhands():
    rank = 13.0
    too_kind = 12.0
    three_kind = 4.0
    two_in_four = 6.0
    full_house = rank * too_kind * three_kind * two_in_four
    return full_house

#Function to calculate 52C5 combination
def choose(n,j):
    Combi = factorial(n) / (factorial(j) * factorial(n - j))
    return Combi

# Main function
def main():
    full = fullhands()
```

```
16  # Main function
17  def main():
18      full = fullhands()
19      total = choose(52, 5)
20      prob = full / total
21      print("Probability of getting a full house: ", prob)
22
23  # Call the main function
24  main()
```


PROBLEMS
OUTPUT
TERMINAL
DEBUG CONSOLE
cmd
Python Deb...
Microsoft Windows [Version 10.0.19045.2486]
(c) Microsoft Corporation. All rights reserved.
C:\Users\zoope\OneDrive\Desktop\STUDENT\UAB\SPRING 2023\CS 355>C:/Users/zoope/anaconda3/Scripts/activate.bat
(base) C:\Users\zoope\OneDrive\Desktop\STUDENT\UAB\SPRING 2023\CS 355> cmd /C "C:\Users\zoope\anaconda3\python.
exe c:\Users\zoope\.vscode\extensions\ms-python.python-2023.2.0\pythonFiles\lib\python\debugpy\adapter/../..\de
bugpy\launcher 62840 -- "C:\Users\zoope\OneDrive\Desktop\STUDENT\UAB\SPRING 2023\CS 355\HW3_Q2.py" "
Probability of getting a full house: 0.0014405762304921968
(base) C:\Users\zoope\OneDrive\Desktop\STUDENT\UAB\SPRING 2023\CS 355>