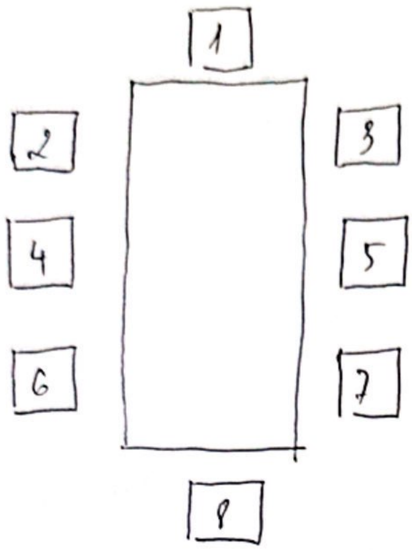
Written Assignment 2

1)



The seating plan :

a) Because Alice is the guest of honour and should sit at the head of the table

=> Alice sit at 1 or 8 => Alice have 2 seats

And 7 another guest can sit anywhere

=> There are $7! = 5040$ different seating arrangements possible

=> There are $5040 \times 2 = 10080$ different arrangements now

b) I will ~~cat~~ ~~cacl~~ caculate how many seating arrangements that Bob and Charlie sit next to or accross the shorter dimension of the table.

+> when Bob sit in 1: Charlie can sit in 2 or 3

+> When Bob sit in 2 : Charlie sit in 3, 1, 4

+> when Bob sit in 3 : 2, 1, 5 (Charlie)

+> When Bob sit in 4 : 2, 5, 6 (Charlie)

+> When Bob sit in 5 : 4, 3, 7 (Charlie)

+> when Bob sit in 6 : 7, 4, 8 (Charlie)

+> When Bob sit in 7 : 6, 5, 8 (Charlie)

+> When Bob sit in 8 : 6, 7 (Charlie)

$\Rightarrow$ We have 22 seats of Charlie

And 6 another guests can sit any where

So $\Rightarrow$ There are $22 . 6!$ seating arrangements when Bob and charlie sit next to or....

$\Rightarrow$ There are $8! - 22.6! = 34.6! = 24480$ seating arrangements sastify this restriction

c) Because Bob and Charlie decided not to turn up.

$\Rightarrow$ There are 6 guests left.

We need to choose 6 seats from 8 seats

And there are C_8^6 ways to do it

And 6 people can sit anywhere in that 6 seats

$\Rightarrow$ There are $C_8^6 . 6! = 20160$ ways.

2)

a) +) There are 4 suits in total, so multiply 4 of the result

+) Each suit has 13 cards, so the number of ways to choose 7 cards from one suit is C_{13}^7

+) The total of ways to choose 7 cards from 52 cards is C_{52}^7

$$\Rightarrow \text{The probability is } \frac{4 \times C_{13}^7}{C_{\Omega}^7} = \frac{4 \times \frac{13!}{7! \times 6!}}{\frac{52!}{7! \times 45!}}$$

$$= \frac{4 \times 13! \times 7! \times 45!}{7! \times 6! \times 52!} = \frac{4 \times 7 \times 8 \times 9 \times 10 \times 11 \times 12 \times 13}{46 \times 47 \times 48 \times 49 \times 50 \times 51 \times 52} = 5,1 \times 10^{-5}$$

Scanned with CamScanner

b) There are 13 possible ranks

$\Rightarrow$ Only 13 ways to choose 4 cards from a rank.

+) Choosing the rank for the 3 of a kind

After choosing the rank, we are left with 12 possible rank.

$\Rightarrow$ There are $C_{12}^{1} \cdot C_{4}^{3} = 12 \cdot 4 = 48$ (ways)

$\Rightarrow$ There ~~624~~. $13 \times 48 = 624$ (ways) to choose a "super - full house"

$$P = \frac{624}{C_{52}^{7}} = \frac{624}{\frac{52!}{7! \cdot 45!}} \approx 4{,}17 \cdot 10^{-6}$$

c) A 3 - pair consists of 2 cards of one rank, 2 cards of seccond rank, 2 cards of third ranks, and one remaining card of a different rank.

+) We need to ~~co~~ choose 3 ranks from 13 ranks

The number of ways is: C_{13}^{3}

+) For each rank, there are 4 cards and we need choose 2 of them

The number is $C_{4}^{2} = 6$

but there 3 ~~three~~ pairs

$\Rightarrow$ So the total ways to do this is: $6^3 = 216$.

+) Choosing the remaining card: There are 10 ranks avaiable, and we need to choose 1 of them $\Rightarrow C_{10}^{1}$ ways.

we have 4 cards in this rank as well

$\Rightarrow C_{10}^{1} \cdot 4 = 40$ ways

So the total ways to choose 3 pair : $40 \cdot 216 \cdot C_{13}^{3}$

$$\Rightarrow P = \frac{40 \cdot 216 \cdot C_{13}^{3}}{C_{52}^{7}} = 1{,}8 \cdot 10^{-2}$$

3)

a) We have these information

+) Probability of the star player being fit for the match: 0,3 => $\boxed{P(\text{not fit}) = 1-0,3 = 0,7}$

+) If the star player play:

-) Win : 0,8

-) Draw : 0,1 (1 - 0,8 - 0,1 = 0,1)

-) Loss : 0,1

+) If the star player does not play ;

-) Win : 0,5

-) Draw : 0,3 (1 - 0,5 - 0,2 = 0,3

-) Loss : 0,2

=> Point if star players is fit :

$$0,3 \times [3 \times 0,8 + 1 \times 0,1 + 0 \times 0,1] = 0,75$$

Point if star players isn't fit :

$$0,7 \times [3 \times 0,5 + 1 \times 0,3 + 0 \times 0,2] = 1,26$$

=> Total point = 1,26 + 0,75 = 2,01 (points)

b) Let's analyze some information:

<u>Strategy 1</u>: +) Star players play in 1st match :

point = $3 \times 0,8 + 1 \times 0,1 = 2,5$

+) 2nd match: we have some

-) If Star players can play :

P (win) = 0,6

P (loss) = 0,1 => Points = $3 \times 0,6 + 1 \times 0,3 = 2,1$

P (draw) = 0,3

-) If star players can't play :

P(win) = 0,3

P (loss) = 0,4 => Points = $3 \times 0,3 + 1 \times 0,3 = 1,2$

P (draw) = 0,3

=> Total point for strategy 1: = 0,5 . (2,1 + 1,2) + 2,5 = 4,15

Strategy 2: The star player did not play the 1st match

+> Similarity to calculate:

Point for 1st match is : ~~6~~ $3 \times 0,5 + 1 \times 0,3$

$= 1,8$ (points)

+) 2nd match:

+> Star player can play : $0,9 \times 2,1 = 1,89$

+> Star can not play : $0,1 \times 1,2 = 0,12$

=> Total point for ~~m~~ 2nd match

$1,89 + 0,12 = 2,01$ (points)

=> Total point of 2 matches :

$2,01 + 1,80 = 3,81$ (points)

=> The best strategy is strategy 1 with 4,15 points

Scanned with CamScanner

first we import the stats package

In [13]:
```
from scipy import stats
```

Then we need to define an appropriate binomial random variable and calculate the probability that more than 10 people will have their tickets rejected

In [15]:
```
# parameters given
n = 850
p = 0.01

# calculate P(X <= 10) and then subtracting it from 1 for results
probability = 1 - stats.binom.cdf(10, n, p)
print(f"The probability that more than 10 people will have their tickets rejected is: {probability:.4f}")
```

```
The probability that more than 10 people will have their tickets rejected is: 0.2358
```

Given problem: Find the largest number of attendees for which the probability that more than 10 tickets are rejected is no more than 0.05.

First, we need to define the function of the probability of more than 10 people are rejected.

In [17]:
```
def more_than_10(n, p):
    return 1 - stats.binom.cdf(10, n, p)
```

I will use a loop to calculate the probability

In [19]:
```
# Define variables
n = 1  # Starting count at 1 attendee
p = 0.01  # Rejection rate

# Initialization to store values
probability = more_than_10(n, p)

# Run the loop with the condition that probability is no more than 0.05
while probability <= 0.05:
  n += 1  # Gradually increase the number of attendees
  probability = more_than_10(n, p)

#We must take 1 away from it and recalculate the probability since n and the probability will both be bigger than the criterion 0.05
```

```python
print(f"The largest number of attendees for which the probability that more than 10 tickets are rejected is no more than 0.05 is: {n}")
```

The largest number of attendees for which the probability that more than 10 tickets are rejected is no more than 0.05 is: 618

In [ ]: I will determine how accurate the test must be so that if

In [21]:

```python
# Define initial value of n and p
n = 850
p = 0.01
tolerance = 0.05
decrement = 1e-4

# Initial probability
probability = more_than_10(n, p)

# Iterate until probability is less than tolerance
while probability > tolerance:
  p -= decrement
  probability = more_than_10(n, p)

print(f"The required rejection rate so that probability is suitable is {p:.4f}")
```

The required rejection rate so that probability is suitable is 0.0072