

Session 7

Question 1

Write a python script to randomly generate a numpy array of size 100 between [1, 6] and print the frequency of each number.

```
In [1]: import numpy as np

# Generate a numpy array of size 100 with random integers between 1 and 6
arr = np.random.randint(1, 7, size=100)
print(arr)

[3 6 4 5 3 5 5 3 1 6 6 6 4 4 2 5 5 1 4 5 2 3 2 3 1 4 1 3 4 6 3 6 5 3 3 3 3
 6 4 1 2 6 3 1 5 2 6 4 3 6 3 6 5 1 1 1 3 5 1 3 1 6 1 3 1 2 6 2 3 4 2 2 4 3
 2 5 6 1 5 1 6 5 3 2 3 6 4 2 3 5 1 6 4 1 5 6 6 3 4 2]
```

```
In [2]: for i in range(1, 7):
    freq = np.where(arr == i)
    print("Frequency of {} is {}".format(i, freq[0].size))

Frequency of 1 is 17
Frequency of 2 is 13
Frequency of 3 is 23
Frequency of 4 is 13
Frequency of 5 is 15
Frequency of 6 is 19
```

Question 2

Write a python script to randomly generate two numpy arrays of size 100 between [1, 6] and print the number of matching rate between them.

```
In [3]: import numpy as np

arr1 = np.random.randint(1, 7, size=100)
arr2 = np.random.randint(1, 7, size=100)

matching = (arr1 == arr2)
rate = np.mean(matching)
print(rate)
```

0.22

Question 3

Write a python script to randomly generate 50 points (x, y) represented as numpy array (50 X 2), then generate another numpy array that represents the distances between that points and the point (0, 0).

Print the points that satisfy the following condition: $\text{distance}(p) \geq \text{average}(\text{distances})$

```
In [4]: import numpy as np

# Generate 50 random points
points = np.random.rand(50, 2)

# Calculate the distances from (0,0)
distances = np.sqrt(np.sum(points**2, axis=1))

# Calculate the average distance
average_distance = np.mean(distances)
print("Distance average {}".format(average_distance))

# Find the points that satisfy the condition
filter = distances >= average_distance
selected_points = points[filter]

print(selected_points)
```

Distance average 0.7397205851125572

```
[[0.86036617 0.72289182]
 [0.87489073 0.4524994 ]
 [0.71713683 0.79609451]
 [0.50070873 0.6556546 ]
 [0.73282182 0.72547949]
 [0.91315412 0.40076449]
 [0.68101846 0.35027044]
 [0.41861798 0.70943886]
 [0.53967201 0.6397804 ]
 [0.74410431 0.08274531]
 [0.8945096 0.93437696]
 [0.47699576 0.63846029]
 [0.90583329 0.66531783]
 [0.03757757 0.94738418]
 [0.77317024 0.5446466 ]
 [0.00215409 0.81073354]
 [0.32224019 0.99012678]
 [0.23157975 0.92642193]
 [0.91136353 0.61669372]
 [0.67804672 0.56671216]
 [0.9408946 0.37485508]
 [0.37825255 0.75191677]
 [0.93576033 0.88638293]
 [0.40903866 0.62677336]
 [0.87916989 0.45410898]
 [0.84652877 0.10268237]]
```

Homewrok

Write a python script enter a circle (center point and radius) and randomly generate 100 points (x, y) represented as numpy array (100 X 2), then print the points that are outside that circle.

```
In [ ]:
```