

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

```
In [2]:  
def my_f(n=1000):  
    h_number=0 #gelen turları saymak için değişken oluşturduk.  
    for i in range(n): #fonksiyona gönderilen n sayısı kadar para atma işlemi yapmak için for döngüsü kullanıyoruz.  
        outcome=np.random.choice(['H','T']) #outcome değişkenine atılan paranın yazı mı tura mı geldiğini random şekilde belirleyerek tanımlıyoruz.  
        if outcome=='H': # para eğer tura geldiyse h_number değişkenini 1 sayı arttırıyoruz.  
            h_number=h_number+1  
    return h_number/n #paranın tura gelme sayısını, para atma işlemi toplamda kaç defa yapıldıysa o sayıya böülüyoruz ve bu değeri geri döndürüyoruz.  
my_f(1000)
```

```
Out[2]: 0.486
```

Paranın yazı ya da tura gelme olasılığı 1/2'dir. Yukarıdaki fonksiyonda fonksiyona gönderilen n değeri kadar para atma işlemi yapılıyor. Daha sonra bu n değeri kadar işleminden paranın Tura geldiği durumlar sayılıyor. Paranın tura geldiği durum sayısı, toplam para atma sayısına bölünüyor. Sonuç olarak 1/2 ye yakın değerlere ulaşılıyor.

```
In [3]:  
def my_f_for_die(n=100):  
    number_of_1=0 #atılan zarın 1 geldiği durumları saymak için değişken oluşturduk.  
    number_of_4=0 #atılan zarın 4 geldiği durumları saymak için değişken oluşturduk.  
    for i in range(n): #fonksiyona gönderilen n sayısı kadar zar atma işlemi yapmak için for döngüsü kullanıyoruz.  
        outcome=np.random.choice([1,2,3,4,5,6]) #Atılan zarın geldiği değeri outcome değişkenine tanımlıyoruz.  
        if outcome==1: #atılan zar eğer 1 e eşit ise number_of_1 değişkenini 1 sayı arttırıyoruz.  
            number_of_1=number_of_1+1  
        if outcome==4: #atılan zar eğer 4 e eşit ise number_of_1 değişkenini 1 sayı arttırıyoruz.  
            number_of_4=number_of_4+1  
    return number_of_1/n,number_of_4/n # atılan zar kaç defa 1 geldiyse ve atılan zar kaç defa 4 geldiyse ayrı ayrı toplamda atılan zar sayısına böülüyoruz ve çırkan değeri döndürüyoruz.  
my_f_for_die(100000)
```

```
Out[3]: (0.16491, 0.16717)
```

1'den 6 ya kadar numaraların olduğu bir zarın bu sayılarından herhangi birinin gelme olasılığı 1/6 yani 0,1666'dır. Yukarıdaki fonksiyonda fonksiyona gönderilen n değeri kadar zar atma işlemi yapılıyor. Daha sonra bu n değeri kadar zar atma işlemi esnasında kaç defa 1 ve 4 geldiği sayılıyor. zarın 1 ve 4 geldiği durum sayıları ayrı ayrı olarak toplam zar atma sayısına bölünüyor. Sonuç olarak 1/6 ya yakın değerlere ulaşılıyor.

```
In [88]:  
def my_f_5(p=True,n=10):  
  
    b_number=0  
    w_number=0  
  
    for i in range(n):  
        c_1=['black','black']  
        c_2=['white','black']  
        c_3=['white','white']  
        cards=[c_1,c_2,c_3]  
        select_card=random.choice(cards) # select_card değişkenine 3 karttan random olarak herhangi birini tanımlıyoruz.  
        if p:  
            print(select_card,end=" ") #tanımlanan kartı ekrana yazdırıyoruz  
  
        select_face=random.choice(select_card) #select_face değişkenine seçilen kartın random olarak herhangi bir rengini ÖnYüz olarak tanımlıyoruz.  
  
        select_card.remove(select_face) #select_card değişkeninden ÖnYüz olarak seçilen kart rengini çıkartıyoruz. Böylece bu değişkende kartın bilinmeyen yani ArkaYüzü kalıyor.  
        unseen_face=select_card[0] #kartın bilinmeyen yani ArkaYüzünü unseen_face değişkenine tanımlıyoruz  
        if p:  
            print("ÖnYüz:",select_face," ArkaYüz:",unseen_face,end=" ") # kartın görünen yüzünü yani ÖnYüzünü(select_face) ve görünmeyen yüzünü yani ArkaYüzünü(unseen_face) yazdırıyoruz.  
  
        if select_face=='black': #kartın görünen yani ÖnYüzü black ise aşağıdaki işlemi gerçekleştirir.  
            b_number=b_number+1  
            if p:  
                print(" ",end=" ")  
            if unseen_face=='white': #kartın görünen yani ÖnYüzü black ve görünmeyen yani ArkaYüzü white ise aşağıdaki işlemi gerçekleştirir  
                w_number=w_number+1  
            if p:
```

```

        print( "+",end=" ")
    if p:
        print()
    else:
        if p:
            print()

return n,b_number,w_number

```

Cekilen kartın görünen yüzünün siyah ve görünmeyen yüzünün de beyaz olma ihtimali üzerine hazırlanan yukarıdaki fonksiyon da siyah-beyaz,beyaz-beyaz ve siyah-siyah olan 3 karttan random seçim yapılıyor. Seçilen kartın görünen yüzünün siyah olduğu durumlar üzerinden ilerlenerek bu durumlar ve görünen yüzünün siyah ve görünmeyen yüzünün beyaz olduğu durumlar bir değişken sayesinde toplanıyor. Fonksiyonda yer alan p değişkeninin True-False olma durumuna göre bu seçilen kart isimleri ekrana yazdırılıyor. Fonksiyon sonlandıında geriye kaç defa bu işlemi yaptığı , kaç defa görünen yüzünün siyah olduğu ve kaç defa görünmeyen yüzünün beyaz geldiği döndürülüyor

In [108...]

```

def my_f_6():
    seen_face_number=0
    unseen_face_number=0
    n=100
    for i in range(n):
        n,b_number,w_number=my_f_5(False,n) #my_f_5 fonksiyonu , fonksiyon içinde yer alan tekrarı n defa olacak şekilde değer gönderiliyor. Buradan dönen değerler eşitliğin sol tarafındaki değişkenlere atanır
        seen_face_number=seen_face_number+b_number # Döngü sonucu defalarca yapılan işlemlerin hepsinde Kartın görünen yüzünün siyah olduğu durumlar toplanıyor.
        unseen_face_number=unseen_face_number+w_number # Döngü sonucu defalarca yapılan işlemlerin hepsinde Kartın görünen yüzünün siyah ve görünmeyen yüzünün beyaz olduğu durumlar toplanıyor.

    return int(seen_face_number/n),int(unseen_face_number/n)

```

my_f_6 fonksiyonu ile my_f_5 fonksiyonuna n değeri gönderilerek my_f_5 fonksiyonu ile kart seçme işlemi n defa yaptırılıyor. Daha sonra bu tüm işlemler sonucu görünen yüzünün siyah ve görünmeyen yüzünün beyaz geldiği durumlar değişkenler sayesinde toplanarak fonksiyona görünen yüz/İşlem sayısı ve Görünmeyen / İşlem sayısı olarak geri döndürüülüyor.

In [114...]

```

for i in range(10): # my_f6() fonksiyonun kaç defa tekrarlanacağını bu döngü ile belirtiyoruz.
    print(my_f_6())

```

```

(49, 16)
(49, 16)
(50, 16)
(48, 16)
(50, 16)
(49, 16)
(49, 16)
(49, 16)
(50, 17)
(50, 16)

```

In []: