TRINAYAN DAS
180123051

In [5]:

```
r=0.0002981060
t=0.0222834

import math
import numpy as np
import random
import matplotlib.pyplot as plt
import statistics

def norm():
  a=random.random()
  b=random.random()
  A,B=2*a-1, 2*b-1
  x=A*A+B*B
  while x>1:
    a=random.random()
    b=random.random()
    A,B=2*a-1, 2*b-1
    x=A*A+B*B
  z=math.sqrt(-2*math.log(x)/x)
  return(z*A)

def lognormal(k):
  sum=0
  for j in range(k):
    m=norm()
    m=r+t*m
    sum=sum+m
  return sum

Price_1=[]
Price_2=[]
for i in range(10000):
  stock=185.4
  d=0.1
  e=math.sqrt(d)
  normal=[]
  price=[]
  poisson=np.random.poisson(0.01,301)
  for j in range(301):
    k=norm()
    normal.append(k)
  for j in range(301):
    k=poisson[j]
    if k!=0:
      z=lognormal(k)
      z=z+(r-(0.5*t*t))*d+t*normal[j]*e
      stock=stock*math.exp(z)
      price.append(stock)

    if k==0:
      z=(r-(0.5*t*t))*d+t*normal[j]*e
      stock=stock*math.exp(z)
      price.append(stock)

  K=1.1*185.4
  s=0
  for j in range(301):
    s=s+price[j]
  l=K-(s/301)
  if l<0:
    l=0
  Price_1.append(l)

  Put_price=K-price[299]
  if Put_price<0:
    Price_2.append(0)
  if Put_price>=0:
    Price_2.append(Put_price)
P_1=[]
for j in range(1000):
  P_1.append(Price_1[j])
A=statistics.mean(P_1)
B=statistics.mean(Price_2)
C=statistics.variance(Price_2)
D=statistics.variance(P_1)
sum=0
for j in range(10000):
  sum+=(Price_1[j]-A)*(Price_2[j]-B)


alpha=sum/(10000*C)

actual=[]
for i in range(1000):
  stock=185.4
  d=0.1
  e=math.sqrt(d)
  normal=[]
  price=[]
  poisson=np.random.poisson(0.01,301)
  for j in range(301):
    k=norm()
    normal.append(k)
  for j in range(301):
    k=poisson[j]
    if k!=0:
      z=lognormal(k)
      z=z+(r-(0.5*t*t))*d+t*normal[j]*e
      stock=stock*math.exp(z)
      price.append(stock)

    if k==0:
      z=(r-(0.5*t*t))*d+t*normal[j]*e
      stock=stock*math.exp(z)
      price.append(stock)

  K=1.1*185.4
  s=0
  for j in range(301):
    s=s+price[j]
  l=K-(s/301)
  if l<0:
    l=0
  Put_price=K-price[299]
  if Put_price<0:
    Put_price=0
  random_variable=l-alpha*(Put_price-B)
  actual.append(random_variable)
```

```
print("The mean and variance respectively of average value Asian put option calc
ulated without using control variate are:", A,D)
print("The mean of the same option by using the price of an European put as the
 control variate :", statistics.mean(actual))
print("The variance of the control variate estimator : " , statistics.variance(a
ctual) )
```

```
The mean and variance respectively of average value Asian put option
calculated without using control variate are: 18.736032528740036 15
9.66627492487626
The mean of the same option by using the price of an European put as
the control variate : 18.573794698674227
The variance of the control variate estimator :  43.82956335858122
```

In [ ]:

```

```