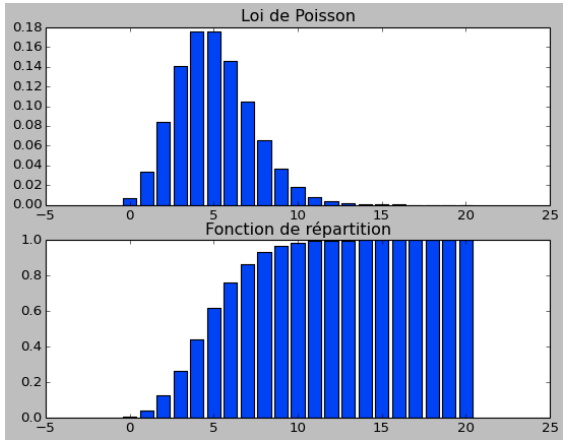
```
## Exercice 1 - Loi de Poisson, tracés
# (1.1)
from math import exp

def factorielle(n):
    P = 1
    for k in range(2,n+1):
        P *= k
    return P


# (1.2)
def loiPoisson(l,k):
    return l**k/factorielle(k)*exp(-l)


# (1.3)
# Paramètres
n = 20
l = 5

# Création des tableaux
K = [k-0.4 for k in range(n+1)] #abscisses
X = [loiPoisson(l,k) for k in range(n+1)]   #loi
Fx = [X[0]] + [0]*n   #Fonction de repartition
for k in range(1,n+1):
    Fx[k] = Fx[k-1]+X[k]

# Tracé
import matplotlib.pyplot as plt
plt.figure(1)
plt.clf()

plt.subplot(2,1,1)
plt.title("Loi de Poisson")
plt.bar(K,X)

plt.subplot(2,1,2)
plt.title("Fonction de répartition")
plt.bar(K,Fx)
plt.show()
```




```
## Exercice 2 - Simulation d'une loi de Poisson
#2.1
from random import random

def simulLoiPoisson(l):
    n = int(10*l**2)+1
    p = l/n
    X = 0
    for loop in range(n):
        if random() <= p:
            X += 1
    return X

#2.2
def simulEspPoisson(l,N):
    S = 0
    for loop in range(N):
        S += simulLoiPoisson(l)
    return S/N

#2.3
def freqLoiPoisson(l,N):
    n = int(10*l**2)+1
```

```
    L = [0] *(n+1)
    for loop in range(N):
        indice = simulLoiPoisson(l)
        L[indice] += 1
    return [x/N for x in L]

#2.4
# parametres
l = 5
N = 1000

# Creation des tableaux
X = freqLoiPoisson(l,N)
K = [k-0.4 for k in range(len(X))]

# Trace
import matplotlib.pyplot as plt
plt.figure(2)
plt.clf()
plt.title("Simulation de la loi de Poisson")
plt.bar(K,X)
plt.show()
```