

T.P. VIII - Variables aléatoires (I)

Code Capytale : 5a40-1016474

Solution de l'exercice 1. [D'après Ecricome - 2016 - Exercice 3]

1.

```
import numpy as np
import matplotlib.pyplot as plt
X = np.arange(0, 23)
U = np.zeros((23, 1))
for k in range(3, 23):
    U[k] = 1/2 * ((4/5)**(k-2) - (2/3)**(k-2))

plt.figure()
plt.plot(X, U, '++')
plt.show()
```

2.

```
F = np.cumsum(U)
plt.figure()
plt.plot(X, F)
plt.show()
```

3. Comme $\mathbf{P}(X \leq m) + \mathbf{P}(X > m) = 1$, on cherche un réel m pour lequel $\mathbf{P}(X \leq m) = 0,5$. On obtient graphiquement $m \simeq 7$. \square

I - Ce qu'il faut savoir

II - Loi uniforme : Exercices

Solution de l'exercice 2. (D'après Ecricome - 2018 - Exercice 3)

```
import numpy.random as rd
def simulation():
    tirage1 = rd.randint(1, 4)
```

```
    if tirage1 < 3:
        res1 = 1
        tirage2 = rd.randint(1, 5)
        if tirage2 == 1:
            res2 = 1
        else:
            res2 = 0
    else:
        res1 = 0
        tirage2 = rd.randint(1, 4)
        if tirage2 < 3:
            res2 = 1
        else:
            res2 = 0
    return res1 + res2
```

\square

Solution de l'exercice 3.

```
import numpy.random as rd

T = 0

if rd.randint(1, 3) == 1:
    for k in range(1, 3):
        if rd.randint(1, 5) < 2:
            T = T + 1
else:
    for k in range(1, 3):
        if rd.randint(1, 5) < 3:
            T = T + 1

print("Une simulation de T : ", T)
```

\square

III - Loïs non uniformes : Exercices

Solution de l'exercice 4.

1.

```
import numpy as np
import numpy.random as rd

A = np.zeros((101, 1))
for k in range(0, 101):
    t = np.randint(1, 5)
    if t <= 2:
        A[k] = 1
    elif t == 3:
        A[k] = 2
    else:
        A[k] = 3

print(A)
```

2. La variable Y contient les valeurs $A[0]$, $A[0] + A[1]$, *ldots*, $A[0] + \dots + A[100]$. Elle contient donc les positions successives de la puce.

Le graphique représente donc la position de la puce au cours du temps.

□

Solution de l'exercice 5. (D'après BCE ESCP - 2020 - Exercice 4)

```
import numpy.random as rd
k = 1 # puce en position k - 1
hasard = rd.randint(1, k+2) # nombre entre 1 et k+1

while hasard <= k: # arrive avec proba k/k+1
    k = k + 1 # se deplace vers la droite
    hasard = rd.randint(1, k+1) # retire au hasard

print("U a pris la valeur :", k)
```

□