DM #3 (facultatif)	*Devrait prendre au moins une dizaine d'heures*	TS4 – pour le 13/05/2019

Exercice 1 : (difficile à partir de d)
On dispose d'une urne contenant 10 boules rouges et 10 boules noires. On effectue *n* tirages successifs avec la règle suivante :
- si on tire une boule rouge, on la remplace par une boule noire
- si on tire une boule noire, on la remplace par une boule rouge.
On se demande quelle sera la répartition des boules au bout d'un grand nombre de tirages.

a) D'après vous, comment évoluera cette répartition ? La situation est-elle la même si on part de 20 boules rouges (et aucune boule noire) ?

b) Simuler deux séries de 100 tirages, soit sur un tableur, soit sur Python (voir page “Comment simuler de l'aléatoire”). Vous enverrez par mail le fichier python (éventuellement lien repl.it) ou le fichier du tableur utilisé (Excel ou LibreOffice).
Vous tracerez le nombre de boules rouges en fonction du nombre de tirages.
Le nombre de boules rouges dans l'urne semble-t-il converger ?

> *Remarque :*
> pour simuler un tirage, on pourra faire comme si les boules étaient toutes numérotées, et que les premières sont toutes rouges.
> Il s'agit donc de prendre un nombre au hasard et de le comparer au nombre de boules rouges.

c) Soit (u_n) une suite telle que $|u_{n+1}-u_n|\geq a$, avec a>0.
Montrer que cette suite ne peut pas être convergente. (On raisonnera par l'absurde en partant de la définition de la limite)
En déduire que le nombre de boules rouges ne peut pas être convergent.

d) On note U_n le nombre de boules rouges au bout de *n* tirages et $V_n=min\{U_k, k\leq n\}$.

- Montrer que (V_n) est décroissante. En déduire qu'elle est convergente.
- Soit *N* fixé. Montrer que, si $U_N=10$ $p(V_{N+10}=0)\geq\frac{10\times9\times...\times2}{20^{10}}$.
- En adaptant ce raisonnement, montrer que, quelle que soit la valeur de U_N , $p(V_{N+20}=0)\geq\frac{20!}{20^{20}}$
- On note $w_n=p(V_{20n}>0)$. Montrer que (w_n) tend vers 0. On pourra :
  - soit utiliser un raisonnement par récurrence et comparer (w_n) à une suite géométrique bien choisie
  - soit montrer que (w_n) est convergente, puis, à l'aide d'un raisonnement par l'absurde, montrer que cette limite ne peut pas être autre chose que 0

e) [Question ouverte] Toutes les répartitions (au bout d'un grand nombre de tirages, par exemple 100 tirages) ont-elles la même probabilité ? La réponde pourra se baser uniquement sur des simulations.

Exercice 2 (difficile à partir de *f* – la *d* n'est difficile qu'à cause des notations, nécessaires pour être rigoureux)
On dispose d'une urne disposant initialement d'une boule rouge et d'une boule noire. On effectue *n* tirages successifs avec les règles suivantes :
- si on tire une boule rouge, on rajoute une boule rouge supplémentaire
- si on tire une boule noire, on rajoute une boule noire supplémentaire.

a) A l'aide de Python ou d'un tableur, simuler 100 tirages. La *proportion* de boules rouges semble-t-elle converger ?
b) On note A_n et B_n le nombre de boules rouges et noires après *n* tirages. On note U_n le couple (A_n, B_n)
Montrer par récurrence que $\forall n\geq0, A_n+B_n=n+2$.
c) Soient a,b>0 fixés. Quelles peuvent être les valeurs de A_{n-1} et de B_{n-1} pour que $A_n=a$ et $B_n=b$?
d) En déduire que $\forall a, b\geq0$, on a :
$$p(U_n=(a,b))=\frac{a-1}{a+b-1}\times p(U_{n-1}=(a-1,b))+\frac{b-1}{a+b-1}\times p(U_{n-1}=(a,b-1)) .$$
e) A l'aide d'un tableur (sans simulation), déterminer la loi de probabilité de A_n (les colonnes correspondront aux différentes valeurs de n, les lignes aux différentes valeurs possibles de A)
Que remarque-t-on ? Démontrer ce résultat par récurrence.
f) On s'intéresse à la proportion de boules X_n au bout d'un très grand nombre de tirages. Soit donc $x\in[0;1]$.
Montrer, à l'aide du résultat précédent, que $p(X_n\leq x)=\frac{E(x(n+1))}{n}$, avec E la fonction “partie entière”.
g) En déduire la limite quand *n* tend vers l'infini de $p(X_n\leq x)$.

<u>Comment simuler de l'aléatoire ?</u>

Pour simuler une répétition de plusieurs expériences, un algorithme est la facon la plus naturelle de procéder. Il est également, pour les cas présents dans ce DM, possible d'utiliser un tableur. Les deux facons sont présentées ici.

1. Python

(si vous n'avez pas Python, vous pouvez télécharger, par exemple, Anaconda à https://www.anaconda.com). Vous pouvez également directement travailler en ligne à https://repl.it mais ce sera moins puissant (particulièrement pour les graphiques)

> *Remarque* : avoir un nombre aléatoire entre 0 et 1 est utile pour simuler un événement dont on connait la probabilité *p*. Il suffit de prendre un tel nombre et de le comparer à *p*. Si c'est plus petit, on considère l'événement réalisé.

Commencer votre code par
```
import random as rnd
```

Pour obtenir un nombre entier aléatoire entre a et b : `rnd.randint(a,b)`
Pour obtenir un nombre réel aléatoire entre 0 et 1 : `rnd.random()`

Exemple : Simuler deux lancers de dés et obtenir la somme
```
def lancers():
      a = rnd.randint(1,6)
      b = rnd.randint(1,6)
      total = a+b
      return total
```

Pour obtenir une représentation graphique, il faut ajouter au début du code :
```
import matplotlib.pyplot as plt
```

Il faut ensuite travailler avec des listes.
plt.hist(L) donnera un histogramme de la liste (effectifs par intervalle, calculé automatiquement. On peut préciser combien on veut d'intervalles : plt.hist(L,50) donnera 50 colonnes.
plt.plot(L) tracera la courbe des termes de la liste, pris dans l'ordre.
plt.show() montrera le graphique tracé.
Remarque : en cas de non mise à jour du graphique, précéder le premier tracé par plt.close()

Exemples :
```
def repetition(n):
      L = []
      for x in range(n):
            resultat = lancers()
            L.append(resultat)
      plt.close()
      plt.hist(L)
      plt.show()
```

Appeler `repetition(50)` donnera l'histogramme de 50 lancers de deux dés.

2. Tableur

Selon la langue du tableur (francais ou en anglais), les instructions suivantes permettent d'avoir des nombres aléatoires :

Entier entre a et b	*Nombre réel entre 0 et 1*
=ALEA.ENTRE.BORNES(a,b)	=ALEA()
=RANDBETWEEN(a,b)	=RAND()