

Exercice 1

1. Soit k, n deux entiers naturels non nuls tels que $k \leq n$. Établir la formule suivante, dite formule du pion :

$$\binom{n}{k} = \frac{n}{k} \binom{n-1}{k-1}$$

Soit $n \in \mathbb{N}^*$. Anthony propose à Bernadette de jouer aux cartes au jeu de tire-pique¹. Le jeu est le suivant : il demande à Bernadette de tirer au hasard n cartes d'un jeu de 52 cartes ($n \leq 52$). Si elle tire l'as de pique, elle gagne. Ce jeu est particulièrement haletant si $n = 52$, mais Anthony choisit $n \neq 52$, car il aime jouer, et, rappelons-le, il est un maître illusionniste.

2. Quelle est la probabilité que Bernadette gagne en fonction de n ?
3. Anthony appelle ce jeu tire-pique parce qu'en fait, il enlève au préalable et au hasard k cartes du paquet ($1 \leq k < n$) à l'insu de Bernadette. Pour que l'illusion soit vraiment trompeuse, Anthony, qui est illusionniste, rappelons-le encore, a conscience du fait que k ne peut pas être égal à 51 par exemple, car Bernadette a des yeux quand même.
 - a) À votre avis (i.e. : pas d'argument attendu), cette filouterie est-elle profitable à Anthony ?
 - b) Vérifiez par le calcul votre intuition en calculant, dans le cas où Anthony pique k cartes au hasard du paquet, la probabilité que Bernadette gagne. Que conseiller à Anthony ?

Exercice 2

Dans un script Python que vous exécuterez, tapez ceci :

```
1 | from random import *
```

1. On considère l'expérience (E) suivante :

(E) : « On tape dans la console la commande suivante : `randrange(0, 2)`.
On observe la réponse de Python »

 - a) Réalisez plusieurs fois l'expérience (E). A-t-on toujours le même résultat ? Comment qualifier l'expérience (E) ?
 - b) Le module `random` permet de simuler des expériences aléatoires. Notamment, la commande `randrange(a, b)` permet de tirer au hasard un entier dans l'ensemble $\{a, \dots, b-1\}$, les nombres a et b étant des entiers. Quels sont alors les résultats observables possibles à l'issue de l'expérience (E) ?
2. On voudrait vérifier que la probabilité d'observer 0 dans l'expérience (E) est de 50%.
 - a) Dans le script précédent, Créer une fonction `def liste_tirage(n)` : qui prend en entrée un entier n et retourne en sortie une liste `resultats` de n termes contenant les résultats de n réalisations consécutives de l'expérience (E).
 - b) Comment interpréter la somme des éléments de la liste `resultats` ?
 - c) Écrire encore dans le script une fonction `def somme(liste)` : qui prend en entrée une liste `liste`, et qui retourne en sortie la somme des éléments de la liste `liste`.

1. Toute ressemblance avec des personnes existant ou ayant existé bla bla bla ...

- d) Vérifiez alors à l'aide de vos fonctions précédentes, et dans la console, que la fréquence observée des 1 en réalisant par exemple 100000 fois l'expérience (E) est voisine de 50%. Indiquez sur votre copie ce que vous avez tapé dans la console et ce que vous avez obtenu.

Remarques. L'expérience (E) simule donc un jeu de pile ou face. On admet que la commande `randrange` fait bien en général ce qu'elle prétend.

3. *Question de probabilités.* Soit $1 \leq p \leq n$ deux entiers naturels. Une urne contient n boules numérotées de 0 à $n - 1$ (indistingables au toucher). On tire au hasard p boules de l'urne. Calculer :
- a) Dans le cas où les p tirages sont simultanés, la probabilité de l'évènement A_1 : «on a tiré les boules numéro 0,1,2,3, ..., $p - 1$ ».
 - b) Dans le cas où les p tirages sont successifs et sans remise la probabilité de l'évènement A_2 : «on a obtenu les entiers 0,1,2,..., $p - 1$ (sans se soucier de l'ordre)».
 - c) Conclusion ?
4. La commande `del(liste[k])` supprime de la liste `liste` son item d'indice k . On souhaiterait simuler un tirage successif sans remise de p entiers parmi les entiers 0, 1, 2, ..., $n - 1$. Compléter les **** dans le script de la fonction suivante qui est censée faire le job :

```

1 def sansRemise(p,n):
2     """ entrée : deux entiers non nuls 1 <= p <= n
3         sortie : une liste de longueur p représentant un
4             tirage au hasard sans remise de p entiers pris dans
5             la liste [0,1,2,...,n-1]
6     """
7     entiers = range(0,n)
8     tirage = []
9     for k in range(p) :
10        r = randrange(0,**)
11        *****
12        tirage.append(*****)
13        del(*****)
14    return tirage

```

5. Anthony aurait-il dû faire des probas avant de faire des tours de magie ? la réponse à cette question est connue dans **Exercice 1 - 3.b)**. Ce résultat, somme toute pas si intuitif, peut être maintenant vérifié expérimentalement avec la fonction `sansRemise`. On représente le jeu de cartes par la liste `range(0,52)`, et on peut supposer sans problèmes que l'as de pique est représenté par l'entier 1. On rappelle que la commande `int(a in L)` retourne 1 si l'objet a est dans la liste L et 0 sinon.
- a) Écrire une fonction `def tirpiq(n,k)` : qui simule une partie de tire-et-pique dans laquelle Bernadette tire n cartes, Anthony en pique k , et retourne 1 si Bernadette gagne et 0 sinon. Pour construire la main de Bernadette, il pourra être judicieux de commencer par construire la liste des numéros des cartes choisies par Bernadette dans le sabot réduit par Anthony.
 - b) Simuler par exemple pour k allant de 2 à 9 et pour $n = 12$, 100000 parties de `tirpiq` et calculer la fréquence des victoires de Bernadette. Confronter cela avec les résultats théoriques obtenus dans **Exercice 1**. Convaincus ?

