Lycée Marcelin BERTHELOT
BCPST 1C – Mme MOREL

Simulation d'expériences aléatoires

1 Tirages dans une liste

Soit `L` une liste qui représente en général une urne, une boîte, etc $\cdots$

1.1 Tirage sans connaître l'indice

La fonction `choice(L)` choisit un élément au hasard dans `L`, selon la loi uniforme.

Remarque 1 Pour un tirage sans remise, il faut enlever de la liste l'élément choisi: `a=choice(L); L.remove(a)`

1.2 Tirage en connaissant l'indice

A noter que l'on peut traiter plus de cas quand on connaît la position de l'élément tiré.

1.2.1 Fonction `randrange()`

$*$ On tire au hasard un indice, selon la syntaxe `k=randrange(len(L))` ou bien `k=randint(0,len(L)-1)`

- $*$ On prend donc l'élément `L[k]`
- $*$ Pour un tirage sans remise, on l'enlève: `L.remove(L[k])`

Vous rappelez-vous de l'inconvénient de la fonction `remove` ?

1.2.2 Fonctions `L.pop()` et `shuffle(L)`

La fonction `L.pop(i)` enlève de la liste `L` l'élément à la ième position. Mais alors, où est le hasard ? Pour cela:

- $*$ La fonction `shuffle(L)` mélange **aléatoirement** les éléments de la liste `L`
- $*$ On prend et on enlève de la liste `L` (ainsi mélangée) son dernier élément : `L.pop()`

Remarque 2 La fonction `shuffle(L)` est très coûteuse en opérations, mais cette méthode est pratique pour tirer plusieurs éléments d'une liste **sans remise**:

```
shuffle(L)
for k in range(n):
   L.pop()
```

Évidemment, n $\leqslant$ len(L)

2 Simulation d'une probabilité

Dans le module `random`, on trouve plusieurs fonctions qui renvoient un nombre aléatoire selon la loi uniforme:

- $*$ `randrange(n)` qui renvoie un nombre aléatoire entre 0 et $n-1$.
- $*$ `randrange(a,b)` qui renvoie un nombre aléatoire entre a et $\boxed{b-1}$ ($a, b \in \mathbb{N}$, et $a < b$).
- $*$ `randint(a,b)` qui renvoie un nombre aléatoire entre a et $\boxed{b}$ ($a, b \in \mathbb{N}$, et $a < b$) .
- $*$ `random()` qui renvoie un réel aléatoire entre 0 et 1.

Proposition 1 Soit p un nombre aléatoire de l'intervalle $[0,1]$ (on note $p \hookrightarrow \mathcal{U}([0,1])$). Alors $\forall \alpha \in [0,1]$, $P(p \leqslant \alpha) = \alpha$.

"preuve":

Application 1 En informatique, pour simuler une probabilité de $\dfrac{1}{3}$ par exemple, on écrira:

```
a=random()
if a <= 1/3 :
   instructions
```

3 Calcul d'une valeur approchée d'une probabilité

Proposition 2 (Loi faible des grands nombres)
Soient X_i des variables indépendantes de Bernoulli de paramètre $P(A)$, où A est un événement de Ω, c'est-à-dire: $X_i(\Omega) = \{0, 1\}$ et $P(X_i = 1) = P(A)$.
Alors on "approche" la probabilité $P(A)$ par la moyenne: $\dfrac{X_1 + X_2 + \ldots + X_m}{m}$, au sens où:

$$\forall \varepsilon > 0\,,\ \lim_{m \to +\infty} P\left(\left|P(A) - \frac{X_1 + X_2 + \ldots + X_m}{m}\right| \geqslant \varepsilon\right) = 0\cdot$$

Remarque 3 Autrement dit, $(X_i = 1)$ ssi A est réalisé donc, si on simule m fois l'expérience aléatoire, alors pour m suffisamment grand ($m \to +\infty$):

$$\boxed{P(A) = \text{fréquence d'apparition de l'événement } A = \frac{\text{nb de fois où } A \text{ est réalisé}}{\text{nb de fois où l'expérience est réalisée}}}$$

Application 2 :

```
def proba(m,args autres):
        c=
        for k in range(  ):
                if "A est réalisé" :
                        ....
        return
```

4 Entraînement

4.1 Expériences aléatoires.

Exercice 1 Pour allumer un feu, on dispose de n allumettes. La probabilité d'allumer le feu avec une allumette donnée est $p \in]0, 1[$.
On considère que le feu est allumé.
Écrire une fonction `Reste` qui simule l'expérience aléatoire et renvoie le nombre d'allumettes restantes.

Exercice 2 On tire avec remise cinq boules d'une urne contenant dix boules numérotées de 1 à 10.
Écrire un programme `python` qui renvoie le maximum des cinq numéros obtenus.

Exercice 3 Soit N un entier supérieur à 10. Une urne contient N boules numérotées de 1 à N. On tire au hasard et simultanément 10 boules de l'urne.
Écrire un programme `python` qui renvoie le minimum des dix numéros obtenus.

4.2 Expériences aléatoires et valeur approchée de probabilité.

Exercice 4 :
Une urne contient n boules numérotées de 1 à n. On tire trois boules de cette urne sans remise (on suppose que $n \geqslant 3$).

1. Écrire une fonction `tirage(n)` qui renvoie le résultat d'un tirage dans une liste, c'est-à-dire la liste des trois boules obtenues.

2. Écrire une fonction `test(n)` qui teste si 2 est le plus petit numéro sorti.

3. Écrire une fonction `proba` qui renvoie une valeur approchée de la probabilité que le plus petit numéro sorti soit égal à 2.

Exercice 5 Lors d'un jeu, un candidat choisit une question en tirant au hasard un papier parmi trois. Il y a:

- une question facile, pour laquelle on a 3 chances sur 4 de donner la réponse exacte,

- une question moyenne, pour laquelle on a 2 chances sur 5 de donner la réponse exacte,

- une question difficile, pour laquelle on a 1 chances sur 5 de donner la réponse exacte.

1. Écrire une fonction `reponse()` qui renvoie `True` si le candidat donne la bonne réponse, `False` sinon.
   on pourra une variable qui vaut `0,1,2` *suivant le niveau de la question.*

2. Écrire une fonction `proba` qui renvoie une valeur approchée de la probabilité de donner la bonne réponse.

4.3 Marches aléatoires et valeur approchée de probabilité.

Exercice 6 Un mobile se déplace aléatoirement le long d'un axe horizontal d'origine 0, sur des points à coordonnées entières, positives ou nulles.
Les déplacements sont effectués selon le protocole suivant:

- à l'instant 0, le mobile est sur l'origine O d'abscisse 0.
- si, pour tout entier naturel n, le mobile se trouve à l'instant n sur le point d'abscisse k ($0 \leqslant k \leqslant n$), alors il sera à l'instant $n+1$, soit sur le point d'abscisse $k+1$ avec la probabilité $\frac{1}{3}$, soit sur le point O avec la probabilité $\frac{2}{3}$.

Ecrire un programme `python` qui simule cette marche aléatoire et renvoie l'abscisse du mobile à l'instant n.

Exercice 7 On dispose d'une pièce et d'un pion placé à l'origine d'un axe gradué. On lance n fois la pièce. A chaque lancer, on déplace le pion d'une unité vers la gauche si on obtient pile, d'une unité vers la droite si on obtient face.
On note X_n l'abscisse du pion à la fin de l'expérience.
Ecrire une fonction `Python` qui simule l'expérience aléatoire et renvoie l'abscisse du pion à la fin de l'expérience.

Exercice 8 Une compagnie aérienne étudie la réservation de l'un de ses vols. Une place donnée est libre le jour de l'ouverture de la réservation et son état évolue chaque jour jusqu'à la fermeture de la réservation de la manière suivante: si la place est réservée le jour k, elle le sera le jour $k+1$ avec la probabilité $\frac{9}{10}$; si la place est libre le jour k, elle sera réservée le jour $k+1$ avec la probabilité $\frac{4}{10}$.
Pour k entier positif, on note r_k la probabilité que la place soit réservée le jour k.

1. Ecrire une fonction `python etat(n)` qui renvoie l'état de la place (réservée ou libre) au jour `n`.
   On pourra prendre une variable qui vaut 1 *si la place est réservée,* 0 *sinon.*

2. Ecrire une fonction `proba(n,m)` qui calcule une valeur approchée de r_n (`m` étant le nombre de fois où on réalise l'expérience)

3. Donner une démarche informatique permettant de conjecturer la limite de la suite (r_n).

Exercice 9 On étudie le mouvement aléatoire d'une puce qui se déplace sur les sommets d'un triangle ABC.
A l'instant 0, la puce est en A et se déplace selon les règles suivantes:

- si à l'instant n, la puce est en A, alors à l'instant $n+1$, elle passe en B avec la probabilité $\frac{1}{3}$ et en C avec la probabilité $\frac{2}{3}$.
- si à l'instant n, la puce est en B, alors à l'instant $n+1$, elle passe en A ou C de façon équiprobable.
- si à l'instant n, la puce est en C, alors à l'instant $n+1$, elle y reste.

1. Ecrire une fonction `marche(n)` qui simule cette marche aléatoire et qui renvoie la position de la puce à l'instant n.

2. On note l'événement A_n ="la puce est en A à l'instant n".
   Ecrire une fonction `proba(n,m)` qui renvoie une valeur approchée de $P(A_n)$. (m est le nombre de fois que l'on réalise l'expérience.)

Exercice 10 Les poules pondent des oeufs que l'on classe suivant trois calibres A, B et C (les petits, les moyens et les gros).

- Si une poule pond un oeuf de calibre A, l'oeuf qu'elle pondra ensuite sera de calibre A, B ou C avec des probablilités respectives de $\frac{1}{2}$, $\frac{1}{4}$ et $\frac{1}{4}$.
- Si une poule pond un oeuf de calibre B, l'oeuf qu'elle pondra ensuite sera de calibre A, B ou C avec des probablilités respectives de $\frac{1}{4}$, $\frac{1}{2}$ et $\frac{1}{4}$.
- Si une poule pond un oeuf de calibre C, l'oeuf qu'elle pondra ensuite sera de calibre A, B ou C avec des probablilités respectives de $\frac{1}{4}$, $\frac{1}{4}$ et $\frac{1}{2}$.

Pour n entier naturel non nul, on désigne par a_n, b_n et $c_n$les probablitités respectives pour que le nième oeuf pondu par une poule soit de calibre A, B ou C.

Ecrire une fonction `python` qui calcule une valeur apporchée de a_n pour tout entier naturel n non nul.