TP INFO n° 11.
Variables aléatoires : quelques activités.

Dans ce TP, vous aurez besoin des bibliothèques `numpy.random` et `matplotlib.pyplot`.

1 Le jeu de Penney

Le jeu de *Penney* est un jeu de pile ou face dont les règles sont les suivantes :

. . . .

- 2 joueurs jouent avec une pièce de monnaie.
- Chaque joueur choisit une séquence de **P** et **F** :

  par exemple **PPF**, **FPP** ou **PFP**. . .

  (ils se sont mis d'accord sur la longueur de la séquence, par exemple 3)

- Ils lancent la pièce et notent les résultats jusqu'à l'apparition de l'une des 2 séquences choisies.

Le joueur ayant choisi cette séquence a gagné !

. . . .

Exercice 1 **Un travail préalable** ■ □ □

En Python, il est plus simple de coder par exemple **Pile** par 0 et **Face** par 1.
On aura donc besoin de savoir faire la conversion (dans les deux sens) entre
- une chaîne de caractère (*ex* : `A='PPFPFPFPPFFFPFP'`),
- une séquence de 0 et 1 (*ex* : `X=[0,1,0,0,1,0,1]`).

1. Ecrire une fonction `chiffres(A)` qui convertit `A` en liste de chiffres.
Exemple : `chiffres('FFPFPFPPFPPP')` renvoie `[1,1,0,1,0,1,0,0,1,0,0,0]`

2. Ecrire une fonction `lettres(X)` qui convertit `X` en chaîne de caractères.
Exemple : `lettres([0,1,1,1,0,1])` renvoie `'PFFFPF'`

Exercice 2 Simulation d'une partie ■■□

Créer une fonction `bataille()` qui simule une partie. Cette fonction doit :
(1) demander au joueur A la séquence qu'il choisit,
(2) demander au joueur B la séquence qu'il choisit,
(3) simuler des lancers de pièce jusqu'à la fin du jeu,
(4) afficher la liste des résultats obtenus sous forme de **P** et de **F**,
(5) Annoncer quel joueur a gagné en affichant : «*le joueur ... a gagne*».

Exercice 3 Simulation de plusieurs parties ■□□

1. Créer une fonction `batailles(N)` qui
(1) simule `N` parties du jeu de Penney,
(2) compte le nombre de parties gagnées par le joueur A,
(3) affiche un message du type :

```
Sur ... parties, la séquence ... a gagné ... fois et la séquence
... a gagné ... fois.
```

2. Observer les phénomènes suivants. On rappelle que la probabilité d'un évènement peut toujours être estimée par la méthode de Monte-Carlo.

(a) **Deux séquences données n'ont pas toujours la même probabilité de victoire**.
Estimer les probabilités de victoire des séquences pour les matchs suivants :

FF contre **FP**, **FF** contre **PF**, **FFFFF** contre **PFFFF**,

Pouvez-vous expliquer les résultats ?

(b) **La relation «... est plus forte que ...» n'est pas transitive.**
Estimer les probabilités de victoire des séquences pour les matchs suivants, et noter les résultats :

PPF contre **PFF**, **PFF** contre **FFP**,
FFP contre **FPP**, **FPP** contre **PPF**.

Interpréter !

- - - - ———————————————— - - - -

Pour d'autres phénomènes surprenants dans ce jeu, et des explications détaillées, si vous êtes intéressé(e) :

http://www.lifl.fr/∼ delahaye/pls/213.pdf

2 Phénomène de dérive génétique : le modèle de Wright-Fisher

Le modèle de Wright-Fisher décrit l'évolution au cours du temps de la fréquence d'un allèle A dans une population. Dans ce modèle, on fait deux hypothèses simplificatrices :

- la population est de taille constante égale à N,
- les générations ne se chevauchent pas.

On note X_k le nombre d'individus portant l'allèle A à la génération k. Si d'aventure à un certain rang on a $X_k = N$ on dit que l'allèle A s'est fixé (de même si $X_k = 0$ l'allèle B s'est fixé).

Règle d'évolution de X_k. Initialement on pose $X_0 = i$ (i est un entier fixé). Les N individus d'une génération $(k+1)$ sont les enfants des N individus de la génération k. On suppose que

- chacun de ces enfants a un parent tiré au hasard parmi les N individus de la génération précédente. Ce tirage se fait uniformément (on n'envisage pas la possibilité d'un avantage sélectif)
- un enfant hérite directement de l'allèle de son parent (on néglige la possibilité de mutations)

Exercice 4 **Simulation de la suite** (X_k) ■■□

1. D'après les hypothèses, quelle est la loi de X_{k+1} sachant que $X_k = a$ (pour $a \in [\![0, N]\!]$) ?

2. Créer une fonction `wrightfisher(i,N)` qui simule la suite (X_k). La simulation s'arrête à la fixation de l'un ou l'autre des allèles. Cette fonction doit

(1) **renvoyer** la liste $[X_0, X_1, \ldots, X_T]$, T étant le numéro de la génération où un des deux allèles s'est fixé,

(2) **afficher** un message du type :

```
Résultat : l'allèle ... s'est fixé à la ... ème génération.
```

3. Amélioration graphique.
Faites tracer à votre fonction `wright_fisher(i,N)` le graphe représentant X_k en fonction de k.

Exercice 5 **Probabilité de fixation de l'allèle A** ■ □ □

Soit p_A la probabilité que l'allèle A finisse par se fixer. Elle dépend bien sûr de i et de N.

1. Créer une fonction `estim_proba(i,N,K)` qui renvoie une valeur approchée de p_A en simulant K fois la suite (X_k). *C'est la méthode de Monte-Carlo.*

2. Avec $N = 50$, tracer la courbe de p_A en fonction de $i \in [\![0, N]\!]$. K sera pris assez grand pour que la courbe obtenue soit à peu près régulière.

3. Recommencer pour d'autres valeurs de N. Conjecturer une expression de p_A en fonction de i et N.

Exercice 6 **Temps de fixation A** ■ □ □

Soit $\mathbb{E}[T]$ le temps moyen de fixation de l'un ou l'autre des allèles. Il dépend bien sûr de i et de N.

1. Créer une fonction `estim_tempsmoyen(i,N,K)` qui renvoie une valeur approchée de $\mathbb{E}[T]$. On pourra simuler K fois l'expérience, puis de faire la moyenne des temps T obtenus.

2. Avec $N = 50$, tracer la courbe de $\mathbb{E}[T]$ en fonction de $i \in [\![0, N]\!]$. K sera pris assez grand pour que la courbe obtenue soit à peu près régulière.

3. Un résultat théorique (issu de calculs complexes) prévoit que pour N grand,

$$\mathbb{E}[T] \simeq -2N\big(x\ln(x) + (1-x)\ln(1-x)\big), \qquad \text{avec } x = \frac{i}{N}.$$

Confronter ce résultat avec la courbe obtenue dans la question 2.

3 L'urne de Pölya

Une urne contient initialement 1 boule rouge et 1 boule blanche. A chaque étape on tire une boule puis on la remet, **en rajoutant une boule de la même couleur**.

On note, pour $n \in \mathbb{N}$:

- X_n le **nombre** de boules blanches dans l'urne après k tirages,
- $P_n = \dfrac{X_n}{n+2}$ la **proportion** de boules blanches dans l'urne après k tirages.

1. Simulation de l'expérience.
Écrire une fonction `simule_polya(N)` qui simule l'expérience et renvoie la liste $[X_0, X_1, X_2, \ldots, X_N]$.

2. Loi de X_n.
Nous avons vu en TD le résultat suivant :

pour tout $n \in \mathbb{N}$, la variable X_n suit la loi uniforme sur $[\![1, n+1]\!]$.

Pouvez-vous illustrer empiriquement ce résultat ? On pourra par exemple présenter sous forme d'histogramme la répartition des valeurs obtenues pour X_n après un grand nombre de simulations de l'expérience.

3. Évolution de la proportion de blanches.
On s'intéresse à l'évolution de la proportion de boules blanches. Écrire une fonction `prop(N)` qui

(1) calcule la liste $[P_0, P_1, \ldots, P_N]$,

(2) représente graphiquement cette liste.

Sur un même graphique, superposer plusieurs réalisations de cette suite. Comment se comporte la suite (P_n) lorsque $n \to +\infty$?

4. Influence des premiers tirages.
Écrire des variantes des fonctions précédentes, avec une composition initiale de l'urne qui est :

- r boules rouges,
- b boules blanches.

Observer l'influence de la composition de l'urne sur les résultats observés précédemment. Essayer notamment en partant de :

- 2 boules de chaque couleur,
- 2 boules blanches et une boule rouge.

4 Le processus de branchement (ou de Galton-Watson

Soit une population à reproduction asexuée qui contient initialement 1 individu (génération 0). On suppose qu'à chaque génération, chacun des individus de la génération, avant de disparaître, donne naissance :

$$\begin{cases} \text{à } \textbf{2 enfants}, \text{ avec probabilité } p \in]0,1[, \\ \text{à } \textbf{aucun enfant}, \text{ avec probabilité } q = 1-p. \end{cases}$$

et que le fait qu'un individu donne ou non naissance à 2 enfants est indépendant des naissances de tous les autres individus de la population.
On remarquera qu'en moyenne, un individu donne naissance à $(2p)$ enfants.

On définit X_n le nombre d'individus que contient la génération n.
On notera p_{ext} la *probabilité d'extinction*, définie par :

$$p_{ext} = \mathbb{P}(\exists n \in \mathbb{N} \ : \ X_n = 0).$$

1. Simulation de l'expérience
Écrire une fonction `simule_branchement(p,x,N)` qui simule puis renvoie la liste $[X_0, X_1, \ldots, X_N]$.

2. Observation empirique.
Demander à votre fonction de représenter graphiquement cette liste. Étudier le comportement de cette suite selon les valeurs de p en observant à chaque fois plusieurs réalisations de la suite (X_n), éventuellement en superposant les graphiques.

3. Probabilité d'extinction.
Nous avons montré en TD, par un calcul théorique, que la probabilité d'extinction vaut :

$$p_{ext} = \begin{cases} q/p, & \text{si } p > 1/2, \\ 1, & \text{si } p \leq 1/2 \end{cases}$$

Observer ce résultat à partir de la simulation.

5 D'autres problèmes, non guidés

• A combien de points se joue un match en n points gagnants ?

Nous avons vu en exercice qu'à l'issue d'un match en n points gagnants, le retard T du perdant avait une espérance $\mathbb{E}(T)$ qui était telle que :

$$\mathbb{E}(T) \underset{n \to +\infty}{\sim} 2\sqrt{\frac{n}{\pi}}$$

Pouvez-vous vérifier expérimentalement ce résultat théorique par simulation ?

• L'urne d'Ehrenfest

Soient deux urnes A et B. Initialement A contient N boules et B n'en contient aucune. A chaque étape du processus on tire au hasard une boule parmi toutes les boules puis on la **change d'urne**.

On note X_k le nombre de boules dans l'urne A après k étapes.

1. Sachant que $X_k = i$, quelle est la loi de X_{k+1} ?

2. Simuler une réalisation de la suite (X_k) jusqu'à un rang T donné. Tracer le graphe de X_k en fonction de k.

3. Comment se comporte cette suite asymptotiquement lorsque N est pris très grand ?