Activité : Paradoxe de Penney

Noms / Prénoms (du groupe) :

Consignes :

Remplir ce document en exploitant le NoteBook Jupyter Paradoxe_de_Penney_Simulations contenant trois programmes Python.

Étape 1 : Introduction à l'expérience aléatoire

1.Qu'est-ce qu'une Expérience Aléatoire (EA) ?

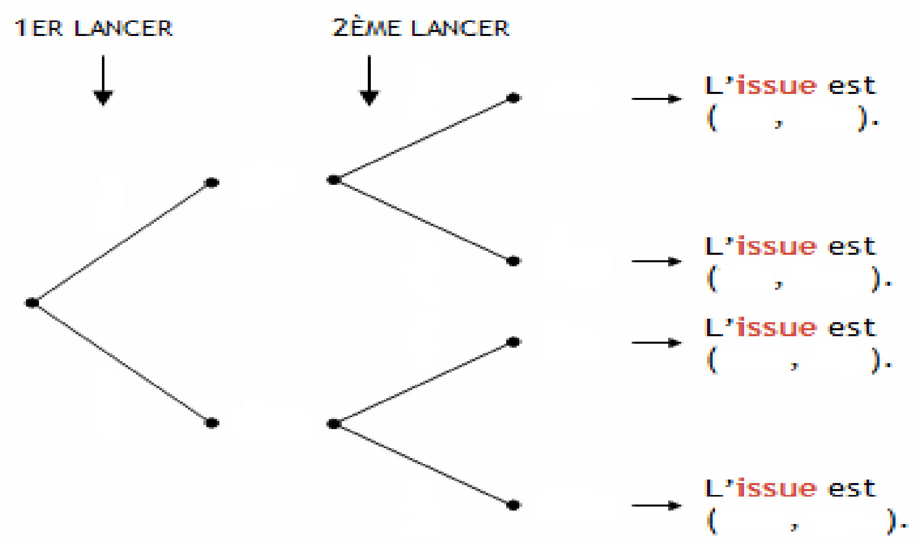
2. Quels sont les résultats possibles lors d'un lancer de pièce ?

3. Quelle est la probabilité d'obtenir Pile (P) ou Face (F) sur un lancer ?

Étape 2 : Étude des séquences avec 2 lancers

4. Quels sont les cas possibles si l'on lance une pièce deux fois ?

5. Compléter l'arbre de probabilité pour représenter tous les cas possibles.



6. Quelle est la probabilité d'obtenir PF ou FF ?

Étape 3 : Rapidité d'apparition entre PF ou FF (2 pièces)

7. Lancer une pièce jusqu'à obtenir la suite PF ou FF. Noter les résultats obtenus ainsi que la suite qui gagne. On fera 10 expérimentations. Que constatez-vous ? (discuter des résultats avec d'autres groupes et votre professeur)

Tirage 1 :	Gagnant :
Tirage 2 :	Gagnant :
Tirage 3 :	Gagnant :
Tirage 4 :	Gagnant :
Tirage 5 :	Gagnant :
Tirage 6 :	Gagnant :
Tirage 7 :	Gagnant :
Tirage 8 :	Gagnant :
Tirage 9 :	Gagnant :
Tirage 10 :	Gagnant :

BILAN (10 lancés) : la suite ..... gagne contre ..... avec une fréquence observée de ....

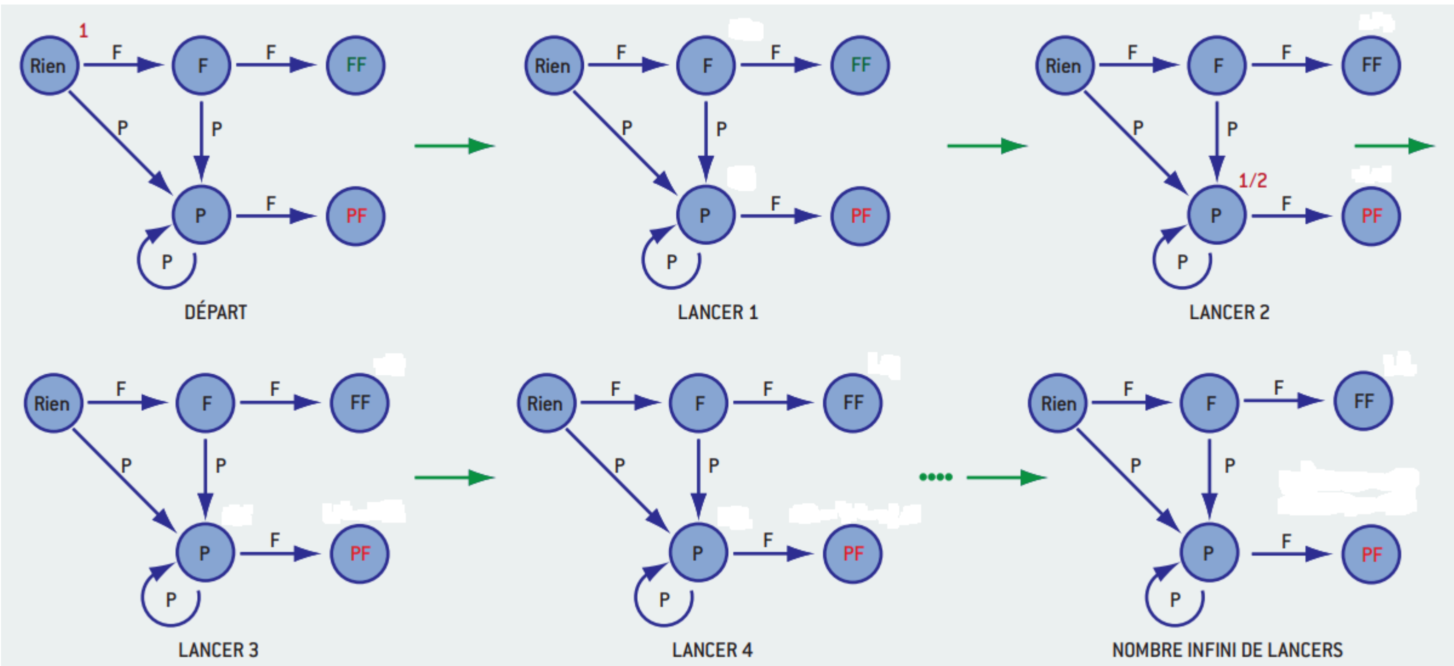
FAITS intéressants observés :

8. Erreur ou réalité ? Pour aller plus loin, il faudrait peut-être faire plus d'essais. Utiliser le programme python “Simulation expérimentale avec 2 lancers” pour générer 100, 1000, 10 000 essais de la bataille PF / FF. Que constate-t-on ?

BILAN (100 lancés) : la suite ….. gagne contre ….. avec une fréquence observée de ……..
BILAN (1000 lancés) : la suite ….. gagne contre ….. avec une fréquence observée de ……..
BILAN (10000 lancés) : la suite ….. gagne contre ….. avec une fréquence observée de ……

CONCLUSION des simulations :

9. (optionnel) Preuve du **Premier Paradoxe de Penney** : en demandant de l'aide à votre professeur, compléter le graphe d'état de la relation d'ordre entre PF et FF pour prouver vos observations.



Pour aller plus loin, on peut se demander si parmi tous les matchs possibles, une suite surclasse toutes les autres … (analyse de 6 matchs).

Étape 4 : Étude des séquences avec 3 lancers

10. Quelles sont toutes les combinaisons possibles avec trois lancers ?

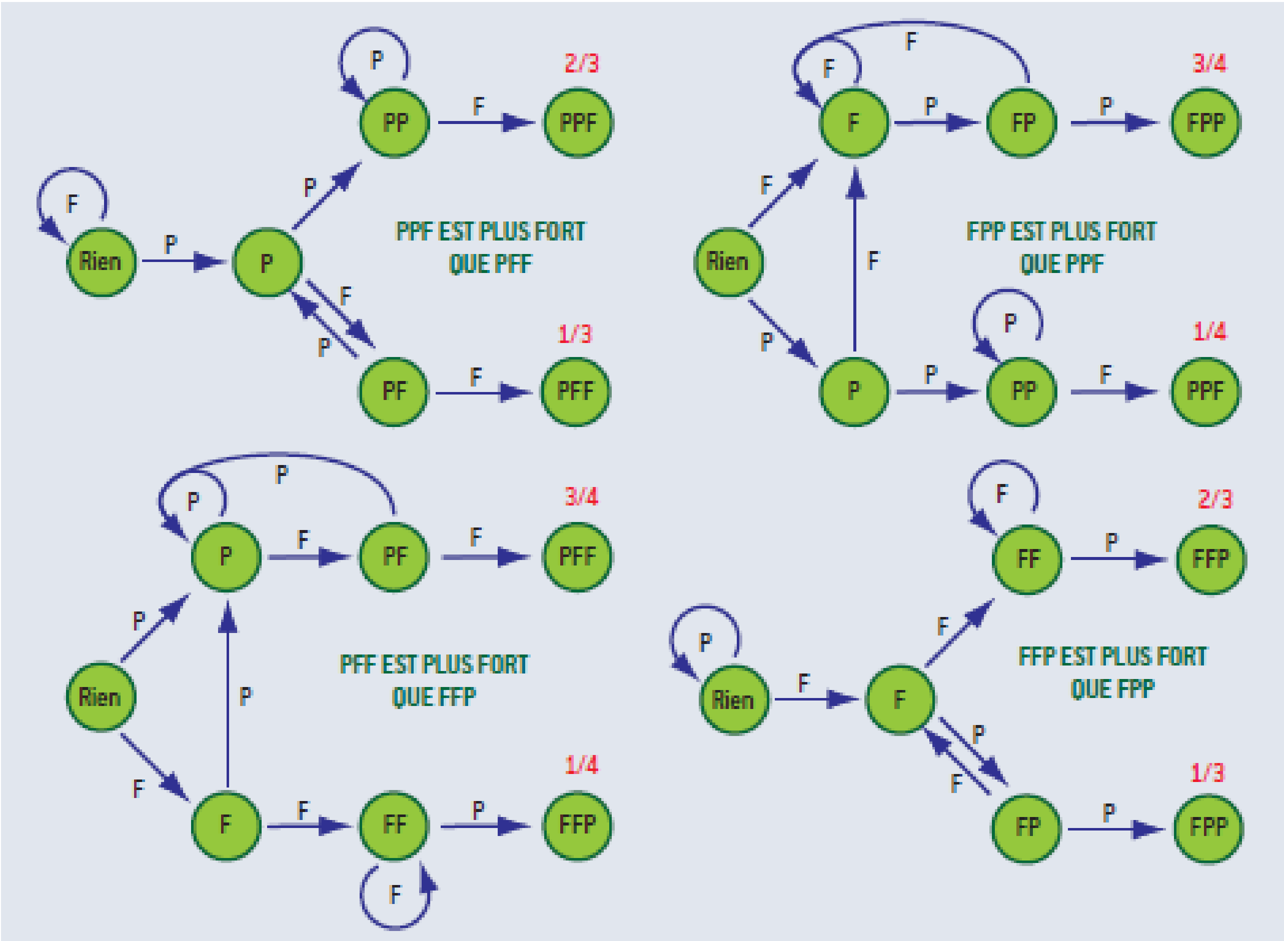
11. Construisez l'arbre des probabilités associé. Que dire des probabilités de chaque lancé de 3 pièces ?

12. Utiliser le programme "Simulation expérimentale avec 3 lancers" pour générer 10000 essais de la bataille FFP / PFF. Qui gagne ?

BILAN (10000 lancés) : la suite ….. gagne contre ….. avec une fréquence observée de ……

CONCLUSION des simulations :

13. Après analyse des quatre graphes d'état, expliquer un **deuxième paradoxe de Penney** (ne pas hésiter à échanger avec d'autres groupes et votre professeur). Cela vous rappelle-t-il un autre jeu très connu ?



Mise en évidence d'une deuxième résultat surprenant :

Étape 5 : Étude des séquences de longueurs différentes

14. Utiliser “Simulation généralisée entre séquences de longueur variable ” pour générer 10000 essais de la bataille PFFF / FFF. Qui gagne ? (**Troisième Paradoxe de Penney)**

BILAN (10000 lancés) : la suite ..... gagne contre ..... avec une fréquence observée de ......

CONCLUSION des simulations :

Étape 6 : Pour aller plus loin …

15. Challenge : trouver une méthode (pas optimale mais) simple pour gagner à chaque fois ?

MÉTHODE SIMPLE PROPOSÉE :

EXEMPLES :

16. Dans la peau d'un génie : comment prévoir automatiquement le résultat - Algorithme de Conway.

4. L'algorithme magique de John Conway

John Conway a conçu un algorithme qui, sans utiliser la méthode des graphes d'états, indique le résultat des confrontations d'une séquence contre une autre et cela quelles que soient leurs complexités et leurs longueurs.

Considérons par exemple les séquences

A = PFPP et **B** = PPFF.

Plaçons **A** au-dessus de **B** :

A PFPP
B PPFF

Si les séquences coïncident, écrivons 1 sur les séquences ; sinon, écrivons 0, comme c'est le cas ici :

0
A PFPP
B PPFF

Comparons maintenant les trois derniers symboles de la première séquence (FPP) avec les trois premiers de la seconde (PPF). Si ces sous-séquences coïncident, on ajoute un 1, sinon un 0 :

00
A PFPP
B PPFF

On compare les deux derniers symboles (PP) de la première séquence avec les deux premiers de la seconde (PP), etc. Nous obtenons finalement :

0011 = 3
A PFPP
B PPFF,

la suite de 0 et de 1 étant lue comme un entier en base 2. C'est la clef de **A** par rapport à **B** ; on la note Clef (**A**, **B**). Ici, Clef (**A**, **B**) = 3. Le rapport de la probabilité que **B** gagne sur la probabilité que **A** gagne est alors donné par :

[Clef (**A**, **A**) – Clef (**A**, **B**)]/[Clef (**B**, **B**) – Clef (**B**, **A**)].

L'application de l'algorithme de Conway donne :

0011 = 3	1001 = 9	0000 = 0	1000 = 8
A PFPP	**A** PFPP	**B** PPFF	**B** PPFF
B PPFF	**A** PFPP	**A** PFPP	**B** PPFF

Clef (**A**, **A**) – Clef (**A**, **B**)) = 6
Clef (**B**, **B**) – Clef (**B**, **A**)) = 8.

Ainsi, **B** = PPFF gagne six fois quand **A** = PFPP gagne huit fois. Autrement dit, la probabilité de gain de **A** contre **B** est : 8/(6 + 8) = 4/7

Autre exemple : **A** = PFP, **B** = PPF.

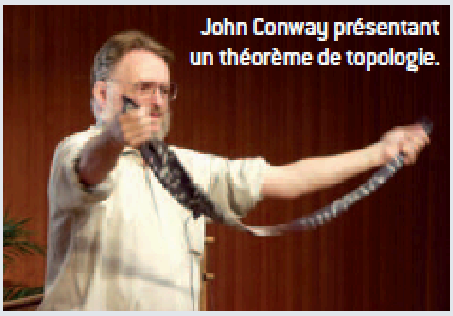
001 = 1	101 = 5	010 = 2	100 = 4
A PFP	**A** PFP	**B** PPF	**B** PPF
B PPF	**A** PFP	**A** PFP	**B** PPF

Clef (**A**, **A**) – Clef (**A**, **B**)) = 4
Clef (**B**, **B**) – Clef (**B**, **A**)) = 2.

Donc **B** = PPF gagne quatre fois quand **A** = PFP gagne deux fois. Dit autrement, la probabilité de gain de **A** contre **B** est : 2/(2 + 4) = 1/3.

Le fait que cet algorithme fonctionne correctement est bien sûr difficile à démontrer, mais cela a été fait de plusieurs manières.

Autre miracle de l'algorithme de Conway : le nombre 2 Clef(**A**, **A**) est le temps moyen d'attente de la séquence **A**. Cela permet de vérifier les résultats de l'encadré 1, qui indiquaient que le temps moyen d'attente de FF est 6 et que celui de PF est 4.


John Conway présentant un théorème de topologie.

Conclusion :

1. Le paradoxe de Penney met en évidence l'importance de la structure des séquences en probabilités.
2. Les arbres de probabilités permettent de visualiser les probabilités de chaque série.
3. Les graphes d'état permettent de mieux appréhender la différence de temps moyen d'apparition lorsque l'on met 2 suites en concurrence.
4. Ces concepts peuvent s'appliquer à des domaines comme la biologie computationnelle, la théorie des jeux ou encore les chaînes de Markov.

Remarque : Les simulations Python sont des outils puissants pour renforcer l'intuition des élèves sur ces phénomènes et leurs enthousiasmes à apprendre.

.