**Lancers d’une pièce**

Introduction à la notion de variable aléatoire

On propose à un joueur de lancer trois fois d’affilée une pièce de monnaie équilibrée.
La règle du jeu est la suivante :

* s’il obtient trois piles, il gagne la partie ;
* s’il obtient deux piles exactement, la partie est nulle ;
* dans tous les autres cas, il perd la partie.
1. Ecrire une fonction Python **piece** qui simule le lancer d’une pièce de monnaie, et renvoie **1** (ou **True**) si le résultat est pile et **0** (ou **False**) sinon.

Aide Python : A l’aide de l’appel au module « **from random import\*** », on peut utiliser l’instruction **randint(a,b)** qui renvoie aléatoirement un entier compris entre a et b.

1. Saisir en console chacune des instructions suivantes, et expliquer leur lien avec la situation étudiée.

**>>>** **lancers=[ piece() for k in range(3) ]**

**>>> lancers**

**>>> sum(lancers)**

1. On considère la fonction Python **jeux** ci-contre, qui reçoit un entier **N** en argument et qui renvoie une liste de trois valeurs.

Indiquer ce que représentent, dans le cadre de l’énoncé, cet entier **N** et les trois valeurs de la liste renvoyée.

L’organisateur du jeu demande une mise de 1€ pour participer au jeu.
Si le joueur gagne, il reçoit 4€, et si la partie est nulle, il est remboursé de sa mise.

1. Ecrire une fonction **gain\_algebrique** qui reçoit en argument une liste **L** (qui sera fournie par la fonction jeux) et qui renvoie le gain algébrique du joueur (le gain algébrique du joueur s’obtient en soustrayant la mise du joueur à la somme qu’il reçoit).
2. Dans cette question, on suppose que **L** est une liste générée par la fonction **jeux**.
3. Que représente le résultat de la saisie **>>> sum(L)** ?
4. Coder et tester la fonction **gm** ci-contre.

Que représente le résultat de la saisie **>>> gm(L)** ?

1. Tester avec 10, 100 puis 10000 parties. Le jeu semble-t-il favorable au joueur ?
2. Calculer les probabilités que le joueur gagne, que la partie soit nulle, puis que le joueur perde. Stocker ces 3 valeurs, dans cet ordre, dans une liste nommée **Proba**.

Calculer la valeur **gm(Proba)** et indiquer la valeur obtenue.

1. **a)** Ecrire une fonction Python **ecart** qui reçoit en argument la liste **L** de répartition des résultats d’une série de parties et renvoie l’écart entre le gain moyen de ces parties et la valeur **gm(Proba)**.

**b)** Tester pour des listes de longueur 100,10000,100000,1000000… Que constate-t-on ?