Exercice 1 : Le problème (ou paradoxe) de Monty-Hall est un célèbre problème de probabilités. Il s'énonce comme suit :

- Un jeu télévisé oppose un présentateur et un joueur.
- Sur le plateau se situent trois portes, l'une cache une voiture et les deux autres cachent une chèvre. Les prix sont répartis par tirage au sort équiprobable. Le présentateur connaît la répartition des prix, mais pas le joueur.
- Le joueur choisit une des trois portes, mais rien n'est révélé.
- Le présentateur ouvre une autre porte (différente de celle choisit par le joueur) ne révélant pas la voiture.
- Le présentateur propose au candidat de changer son choix de porte à ouvrir définitivement.

On se propose de comparer deux stratégies pour le joueur :

- Stratégie 1 : toujours rester sur le choix initial
- Stratégie 2 : toujours changer de choix

Sur internet, les débats entre amateurs de mathématiques font rage. Certains soutiennent que les deux stratégies se valent tandis que d'autres affirment que l'une donne plus de chances de gagner la voiture.

Pour essayer de tirer cela au clair, on réalise une simulation informatique de l'expérience. On considère les deux fonctions **Python** qui suivent. On rappelle que l'appel `j % 3` renvoie le reste de la division euclidienne de l'entier j par 3.

```python
def mystere1():
    S = 0
    for k in range(10**6):
        listePortes = [0,1,2]
        voiture = rd.randint(0,3)
        choixJoueur = rd.randint(0,3)
        j = (choixJoueur + 1) % 3
        if voiture == j:
            j = (j+1) % 3
        del listePortes[j]
        if choixJoueur == voiture:
            S = S + 1
    return S / 10**6
```

```python
def mystere2():
    F = 0
    for k in range(10**6):
        listePortes = [0,1,2]
        voiture = rd.randint(0,3)
        choixJoueur = rd.randint(0,3)
        j = (choixJoueur + 1) % 3
        if voiture == j:
            j = (j+1) % 3
        del listePortes[j]
        choixJoueur = (choixJoueur + 1) % 3
        if not (choixJoueur in listePortes):
            choixJoueur = (choixJoueur + 1) % 3
        if choixJoueur == voiture:
            F = F + 1/10**6
    return F
```

1. Décrire (en expliquant) ce que renvoie l'appel `mystere1()`.

Démonstration. Dans ce programme, le joueur fait son premier choix ligne 6, puis ce choix ne change plus jusqu'à-ce qu'on teste si il a trouvé la voiture (ligne 11). Ainsi, ce programme simule l'expérience avec la stratégie 1.
Ce programme renvoie la fréquence empirique de réalisation de l'événement « le joueur gagne la voiture » sur 10^6 simulations. Ainsi, d'après la loi faible des grands nombres, il renvoie une approximation de la probabilité de l'événement « le joueur gagne la voiture » lors de l'utilisation de la stratégie 1. □

2. Décrire (en expliquant) ce que renvoie l'appel `mystere2()`.

Démonstration. Dans ce programme, le joueur fait son premier choix ligne 6, puis les lignes 11 à 13 permettent de simuler le changement de porte de la part du joueur. Les lignes 7 à 10 simulent l'ouverture d'une porte perdante par le présentateur. Ainsi, ce programme simule l'expérience avec la stratégie 2.
Ce programme renvoie la fréquence empirique de réalisation de l'événement « le joueur gagne la voiture » sur 10^6 simulations. Ainsi, d'après la loi faible des grands nombres, il renvoie une approximation de la probabilité de l'événement « le joueur gagne la voiture » lors de l'utilisation de la stratégie 2. □

3. L'appel `mystere1()` renvoie 0.333323 tandis que l'appel `mystere2()` renvoie 0.666474. Quelle stratégie choisiriez vous si vous étiez invité·e à jouer à ce jeu télévisé ?

Démonstration. D'après les simulations **Python**, le joueur a environ 2 chances sur 3 de gagner la voiture avec la stratégie 2, tandis qu'il a 1 chance sur 3 de la gagner avec la stratégie 1. Puisque $\frac{2}{3} > \frac{1}{3}$, je choisirais de jouer avec la stratégie 2 (toujours changer de porte). □

4. Retrouver le résultat obtenu par simulation informatique via un argument simple de dénombrement.

Démonstration.
- Avec la stratégie 1, le joueur à une chance sur trois de tomber sur la voiture lorsqu'il fait son premier choix. De plus, $\frac{1}{3} \simeq 0.333323$.
- Avec la stratégie 2, le joueur à toujours une chance sur trois de tomber sur la voiture lorsqu'il fait son premier choix. Ceci se reformule en disant que la voiture à deux chances sur trois d'être derrière l'une des deux portes non choisies. Le présentateur sait où elle se trouve et ouvre toujours une porte avec une chèvre. Ainsi, après que le présentateur ait ouvert une porte, la voiture à deux chances sur trois d'être derrière la porte non choisie par le joueur et non ouverte par le présentateur. Si le joueur change de porte, il aura donc deux chances sur trois de gagner la voiture. De plus, $\frac{2}{3} = 0.666474$.

□