

TP n°4 (exercice supplémentaire)

Exercice 1. Dans les années 1960, un jeu télévisé américain proposait au participant de partir avec ce qu'il trouvait derrière une porte. Dans ce jeu, le candidat est face à trois portes. Il sait que derrière l'une de ces portes est cachée une **voiture** et derrière les deux autres, une **chèvre**.

Il choisit une porte sans l'ouvrir. Alors, le présentateur, Monty Hall, qui sait où se cache la voiture, ouvre l'une des deux portes derrière laquelle se trouve une chèvre. Il demande ensuite au candidat de **conserver son choix initial** ou bien de **changer de porte**.

Le but de cet exercice est de déterminer la meilleure stratégie à adopter du point de vue des probabilités.

Voici un script qui servira à répondre aux questions de la première partie.

```

from random import randint
voiture=randint(1,3)
choix=randint(1,3)
if choix==voiture :
    gagne_changeant =0
    gagne_sans_changer=1
else :
    gagne_changeant=1
    gagne_sans_changer=0
print("s'il change de porte :")
if gagne_changeant ==1 :
    print("le candidat gagne")
else :
    print("le candidat perd")
print("s'il garde sa porte:")
if gagne_changeant == 1 :
    print("le candidat perd")
else :
    print("le candidat gagne")

```

1. Partie 1 :

- Quelle est la signification de la ligne 1 ?
- Expliquer la ligne 2. A quoi correspondent les variables *voiture* et *choix* ? Quel peut-être la signification de la commande *randint* ?
- Ouvrez un nouveau fichier et enregistrez-le dans votre dossier personnel sous le nom **TP4bis NOM**.
- Copier le programme dans l'**éditeur** et l'exécuter plusieurs fois (environ 10 fois par exemple). Conjecturer la meilleure stratégie entre *changer de porte* ou *garder la porte initiale*.

2. **Partie 2** : afin d'automatiser le dénombrement des parties, nous décidons de modifier le programme précédent afin qu'il renvoie la fréquence (pour n tentatives) de parties gagnées en *changeant de porte* et la fréquence de partie gagnées *sans changer de porte*.

```
from random import randint
def jeu(n) :
    garde =0
    change=0
    for i in range (?) :
        voiture=randint(0,2)
        choix=randint(0,2)
        if choix == ? :
            ouverte=(voiture+1+randint(0,1))%3
            garde=garde+1
        else :
            ouvert=3-choix-voiture
            change= ?
    return (garde/n, change/n)
```

- Compléter la fonction ci-dessus (en remplaçant les points d'interrogations) afin qu'elle réponde au problème posé. *Indication* : un travail préalable, au brouillon, avec des exemples permettra de mieux comprendre le programme.
- Expliquer la ligne 5. Comment la traduiriez-vous en langage *Scratch* ?
- Expliquer la ligne 12. *Indication* : cette commande correspond à une action du présentateur.
- Copier cette fonction dans **l'éditeur**, déterminer les fréquences pour $n = 1000$ parties jouées et conjecturer la meilleure stratégie.