

Activité : le paradoxe de Saint-Petersbourg

On considère le jeu de « Pile ou face » suivant :

Le joueur gagne le double de sa mise en cas de victoire (« Pile »), et perd sa mise en cas de défaite (« Face »).
Le joueur peut rejouer tant que sa réserve d'argent le lui permet.

Pierre, qui dispose d'une réserve de 1000€, décide de jouer de la manière suivante :

- Il mise 1€ au départ du jeu ;
- Tant qu'il perd, il rejoue une partie en doublant sa mise ;
- Il arrête si il gagne, ou si il n'a plus assez d'argent.

1. Que se passe-t'il si Pierre joue et fait Face-Face-Face-Pile ?
2. Afin de simuler une partie, on propose la fonction (incomplète) ci-dessous :

```
def jeu():
    from random import randint
    victoire = False
    mise = 1
    reserve = 1000
    while ...:
        reserve = reserve - mise
        tirage = randint(0,1)
        if tirage == 0:
            reserve = reserve + 2 * mise
            victoire = True
        else:
            mise = mise * 2
    return reserve - 1000
```

Retrouver, dans l'énoncé, la condition à vérifier dans la boucle `while` de la fonction.
Recopier et compléter la fonction dans spyder.

3. Que représente la valeur renvoyée par la fonction ?
4. Écrire un programme qui simule l'exécution de n parties, n étant un nombre choisi par l'utilisateur, et qui renvoie la fréquence de parties avec un gain positif d'argent.
5. Effectuer plusieurs simulations afin de conjecturer une valeur approchée de la probabilité de gagner de l'argent.

Boîte à outils

- L'instruction conditionnelle Si « condition » Alors « instruction » Sinon « instruction » s'écrit de la manière suivante.

```
if condition:
    instructions
else:
    instructions
```

- La boucle Pour s'écrit de la manière suivante. Boucle avec un compteur k variant de a à b .

```
for k in range (a,b+1):
    instructions
```

MÉMENTO PYTHON : VOIR RABATS

- La boucle Tant que s'écrit de la manière suivante.

```
while condition:
    instructions
```

- Pour simuler le choix d'un nombre aléatoire entre 0 et 1, on utilise la fonction `random()`
- Pour simuler le choix d'un nombre aléatoire entier entre a et b , on utilise la fonction `randint(a,b)`
- Pour tester si deux nombres entiers a et b sont égaux, la condition s'écrit : `a==b`
- Pour tester si deux nombres entiers a et b ne sont pas égaux, la condition s'écrit : `a!=b`