

Objectif : cette séance a pour but de vous faire programmer en Python à l'aide de la bibliothèque Random. Dans un premier temps vous allez modéliser le lancer d'une pièce équilibrée et celui d'une pièce truquée. Dans un second temps vous regarderez si les fréquences d'apparitions coïncident avec vos calculs théoriques. Enfin vous répéterez le lancer de deux pièces en étudiant les résultats possibles et les fréquences d'apparitions des différentes issues.

I Une pièce équilibrée VS une pièce truquée :

1. Ouvrir le logiciel "Pyzo" et créer un nouveau fichier qu'il faut sauvegarder sur le bureau avec l'intitulé suivant : "Nom1_Nom2_lancer_pieces.py"
2. Importer la bibliothèque "Random" comme "rd" dans votre programme "Python".
3. Recopier et remplir le code ci-dessous pour que la fonction "lancer_une_piece_equilibree()" modélise le lancer d'une pièce équilibrée.

```
1 def .....():
2     k = rd.randint(...,...)
3     return k
```

4. Recopier et remplir le code ci-dessous pour que la fonction "lancer_une_piece_truquee()" modélise le lancer d'une pièce truquée qui donne 4 fois plus de "Face" que de "Pile".

```
1 def .....():
2     k = rd.randint(...,...)
3     if k <= .....:
4         return ....
5     else:
6         return ....
```

5. On note **X1** la valeur de la pièce équilibrée et on note **X2** la valeur de la pièce truquée. Donner la loi de l'expérience aléatoire associée à **X1** et donner la loi de l'expérience aléatoire associée à **X2** dans un même tableau.

Définition 1. Soient $n \in \mathbb{N}$, on considère une expérience aléatoire et une issue associée à cette expérience. On répète n fois l'expérience et on compte le nombre de fois que l'issue voulue apparaît à l'aide d'un "compteur". La fréquence F d'apparition de cette issue est :

$$F = \frac{\text{compteur}}{n}$$

6. (a) Si on répète un grand nombre de fois le lancer d'une pièce équilibrée et celui d'une pièce truquée et que l'on compte le nombre de "Piles" observé, comment évolueront les fréquences d'observation de l'issue "Pile" plus on répète ces expériences aléatoires ?
- (b) Recopier et remplir le code qui est sur la page 2 pour que la fonction "frequence(n)" nous renvoie les fréquences d'observation de l'issue "Pile" pour n lancers de pièces équilibrées et n lancers de pièces truquées.
- (c) À l'aide de la fonction "frequence(...)" remplir le tableau sur la page 2 pour donner les fréquences d'observation de l'issue "Pile" pour les différents nombres de lancers.

```

1 def .....(...): #n est le nombre de lancers
2     compteur1 = 0
3     compteur2 = ....
4     for i in range(...):
5         a = lancer_une_piece_equilibree()
6         b = .....()
7         if a ==....:
8             compteur1 = compteur1 + ....
9         if b .... :
10            compteur2 = ..... + ....
11     frequence1 = ...../n
12     frequence2 = ...../....
13     return frequence1,frequence2

```

	2	5	10	20	50	100	1000	10000
F1								
F2								

II Répétition de deux lancers avec une pièce équilibrée :

1. Recopier le code ci-dessous pour que la fonction "*repetition_2_lancers_equilibrees()*" modélise le lancer de deux pièces équilibrées. Donner les issues possibles associées à cette expérience aléatoire.

```

1 def repetition_2_lancers_equilibrees():
2     p1 = lancer_une_piece_equilibree()
3     p2 = lancer_une_piece_equilibree()
4     somme = p1+p2
5     return somme

```

2. Recopier et remplir le code ci-dessous pour que la fonction "*frequence_2_pieces_equilibrees(...)*" donne les fréquences d'observation des issues associées aux lancers de deux pièces équilibrées.

```

1 def frequence_2_pieces_equilibrees(n):
2     compteur0 = 0
3     compteur1= ...
4     compteur2= ...
5     for i in range(n):
6         a= repetition_2_lancers_equilibrees()
7         if a == .... :
8             compteur0 = compteur0 + 1
9         elif .... == ... :
10            compteur1 = ..... + ...
11         else:
12            compteur2 = ..... + ...
13     frequence0 = compteur0/n
14     frequence1 = ...../n
15     frequence2 = ...../...
16     return [frequence0, frequence1,frequence2]

```

3. À l'aide de la fonction "*frequence_2_pieces_equilibrees(...)*" remplir le tableau.

	2	5	10	20	50	100	1000	10000
F0								
F1								
F2								

III Répétition de deux lancers avec une pièce truquée :

1. Écrire le code d'une fonction *"repetition_2_lancers_truquees()"* pour qu'elle modélise le lancer de deux pièces truquées. Donner les issues possibles associées à cette expérience aléatoire.
2. Recopier et remplir le code ci-dessous pour que la fonction *"frequence_2_pieces_truquees(...)"* donne les fréquences d'observation des issues associées aux lancers de deux pièces truquées.

```
1  def frequence_2_pieces_truquees(n):
2      compteur0 = 0
3      compteur1= ...
4      compteur2= ...
5      for i in range(n):
6          a= .....()
7          if a == .... :
8              compteur0 = ..... + ...
9          elif .... == ... :
10             compteur1 = ..... + ...
11         else:
12             compteur2 = ..... + ...
13         frequence0 = compteur0/n
14         frequence1 = ...../n
15         frequence2 = ...../...
16     return [frequence0, frequence1, frequence2]
```

3. À l'aide de la fonction *"frequence_2_pieces_truquees(...)"* remplir le tableau.

	2	5	10	20	50	100	1000	10000
F0								
F1								
F2								

IV Répétition de deux lancers avec une pièce truquée :

1. Avec ce que vous venez de faire précédemment, donner une conjecture sur la loi de probabilité associée expériences : le lancer de deux pièces équilibrées, le lancer de deux pièces truquées.
2. Faire un arbre de probabilité qui représente le lancer de deux pièces équilibrées et en déduire la loi de cette expérience aléatoire.
3. Avec ce que vous venez de faire précédemment, donner une conjecture sur la loi de probabilité associée expériences : le lancer de deux pièces truquées.
4. Faire un arbre de probabilité qui représente le lancer de deux pièces truquées et en déduire la loi de cette expérience aléatoire.