IC Devoir 7 - 1h - vendredi 6 juin

Exercice 1 Donner une fonction récursive, `det(M)`, qui renvoie le déterminant de la matrice `M` de type `np.array`, selon le développement par rapport à une ligne.
Rappel de la formule : pour $M \in \mathcal{M}_n(\mathbb{R})$ et $i \in [\![1,n]\!]$

$$\det(M) = \sum_{k=1}^{n} m_{i,k}(-1)^{i+k}\Delta_{i,k}$$

Exercice 2 Considérons un ensemble donné sous la forme d'une liste `E`. Dans chaque cas, identifier "l'objet" simulé et donner quelques explications :

simulation 1

```
def f(E,p):
    S=[]
    for i in range(p):
        r=rd.randint(len(E))
        S.append(E[r])
    return S
```

simulation 2

```
def g(E,p):
    assert p<=len(E),'ENC'
    S=[]
    for i in range(p):
        r=rd.randint(len(E))
        S.append(E[r])
        E.pop(r)
    return S
```

simulation 3

```
def h(E,p):
    assert p<=len(E),'ENC'
    S=[]
    for i in range(p):
        r=rd.randint(len(E))
        S.append(E[r])
        E.pop(r)
    S.sort()
    return S
```

Exercice 3 *Lancers d'une pièce*
On effectue une succession infinie de lancers indépendants d'une pièce donnant Pile avec la probabilité $p \in]0,1[$ et Face avec la probabilité $q = 1-p$.
On dit que la première série est de longueur $n \geq 1$ si les n premiers lancers ont amené le même côté de la pièce et le $(n+1)$-ième l'autre côté.
De même la deuxième série commence au lancer suivant la fin de la première série et se termine (si elle se termine) au lancer précédant un changement de côté.

1. Compléter la fonction suivante qui donne la longueur des deux premières séries :

```
def longeurs(p):
    L1,r=1,rd.rand()<p
    s=...
    while r==s:
        L1=...
        s=...
    L2=...
    r=rd.rand()<p
    while ...:
        L2=...
        ...
    return L1,L2
```

2. Donner une fonction `series(n,p)` qui simule l'expérience et retourne les longueurs de n premières séries. ❄❄

Exercice 4 Arcsin - *Intégration numérique*
1. Présenter brièvement la méthode des rectangles qui donne une approximation de $\int_a^b f(t)\mathrm{d}t$.
En particulier, vous complèterez la formule suivante avec n le nombre de subdivisions :

$$\int_a^b f(t)\mathrm{d}t \approx \frac{b-a}{n}\sum_{k=0}^{n-1}\cdots$$

2. On rappelle que $\mathrm{Arcsin}'(x) = \dfrac{1}{\sqrt{1-x^2}}$.
Compléter le script d'une fonction `arcsin(x,h=1e-5)` qui retourne une approximation de $\mathrm{Arcsin}(x)$ où h est le pas de la subdivision initialisé par défaut à 10^{-5} :

```
def arcsin(x,h=1e-5):
    if x<0:
        h=-h
    n=int(x/h)
    t,I=...
    for i in range(1,n):
        t=...
        I=I+1/np.sqrt(1-t**2)
    return ...
```

Exercice 5 *Modèle de File non bornée*
Considérant une file non bornée (FIFO : first input, first output), modélisée par une liste.

- Les éléments sont rangés dans l'ordre où ils ont été empilés.
- La longueur de la liste correspond à la hauteur de la pile.

Définir les fonctions de gestion d'une file.

1. `creer_file()` : renvoie une file vide
2. `defiler(f)` : enlever un élément de la file et le renvoie
3. `enfiler(f,v)` : ajouter un élément de la file
4. `taille(f)` : renvoie la taille de la file
5. `est_vide(p)` : teste si la file est vide

Lycée Saint-Exupery – maffre.saintexupery@gmail.com – MPSI