DS n° 04

- Faire tous les exercices dans un même fichier NomPrenom.py à sauvegarder,
- mettre en commentaire l'exercice et la question traités (ex : # Exercice 1),
- ne pas oublier pas de commenter ce qui est fait dans votre code (ex : # Je crée une fonction pour calculer la racine d'un nombre),
- il est possible de demander un déblocage pour une question marquée *, mais celle-ci sera notée 0,
- il faut vérifier avant de partir que le code peut s'exécuter et qu'il affiche les résultats que vous attendez.

Exercice 1 : Longueur de ligne brisée et lecture d'un fichier

Le tableau suivant donne les coordonnées d'un point $M(t)$ (abscisse $x(t)$ et ordonnée $y(t)$) en divers instants t :

t	0.1	0.2	0.3	0.4	0.5
$x(t)$	1	0	-1	0	1
$y(t)$	0	1	0	-1	0

1 * Définir deux listes `LX` et `LY` contenant les abscisses et les ordonnées du point $M(t)$ aux divers instants t.

2 Représenter les points aux divers instants t ainsi que la ligne polygonale joignant ces points.
On utilisera `plot` et `show` du module `matplotlib.pyplot`.

3 Calculer et afficher la longueur de cette ligne polygonale.

Les coordonnées du point $M(t)$ sont maintenant stockées dans un fichier nommé num-points.csv situé dans le répertoire GNA. Chaque ligne de ce fichier est constituée des données "$(t, x(t), y(t))$" associées à un point $M(t)$. Le séparateur de colonnes est ici ','.

4 * A partir de ce fichier, définir trois listes de flottants `LT`, `LX` et `LY` contenant les instants t, les abscisses et les ordonnées du point $M(t)$.

5 Vérifier que la liste `LT` est bien ordonnée selon ses valeurs croissantes.

6 Calculer et afficher la longueur de cette ligne polygonale.

7 Afficher la vitesse moyenne sur l'ensemble du parcours.

Exercice 2

La suite $(K_n)_{n\in\mathbb{N}}$ d'entiers naturels est définie par :

- $K_0 = 1$
- $\forall n \in \mathbb{N}^*,\ K_n = \sum_{i=0}^{n-1} K_i K_{n-1-i}$

1 * Écrire une fonction récursive `K` d'argument un entier n et renvoyant K_n.

2 Calculer et afficher `K(2)`, `K(5)`, `K(10)` et `K(15)`. Que pensez-vous de l'efficacité de la fonction `K` ? Réponse en commentaire.

3 * Écrire une fonction `fact` **(non récursive)** qui prend comme argument un entier n positif et qui renvoie la valeur de la factorielle : $n!$.

4 Écrire une fonction `LK` non récursive d'argument un entier n et renvoyant le nombre L_n donné par

$$L_n = \frac{1}{n+1}\binom{2n}{n}.$$

Rappel : $\binom{n}{k} = \dfrac{n!}{k!(n-k)!}$

5 Afficher en les comparant les 10 premières valeurs de K_n et de L_n.

6 Que peut-on conjecturer ? Réponse en commentaire.

Exercice 3 : tri dans le plan

On considère le code suivant :

```
def ordonner(L) :
    if len(L) <= 1 :
        return L
    e0 = L[0]
    n = 1
    Linf, Lsup = [], []
    for ei in L[1:]:
        if ei == e0 :
            n += 1
        elif ei < e0 :
            Linf.append(ei)
        else :
            Lsup.append(ei)
    return ordonner(Linf)+n*[e0]+ordonner(Lsup)
```

1 La fonction `ordonner` est une fonction de tri récursive. Pour la liste $L = [5, 3, 10, 7, 7, 2]$, comptez (à la main) le nombre de fois où la fonction `ordonner` sera appelée. Réponse en commentaire.

2 Proposer et écrire une autre fonction de tri.

3 * Il n'y pas de relation d'ordre intéressante dans $\mathbb{R}^2$. On peut néanmoins construire une relation de comparaison. On dira que "$L1 = [a, b]$ est inférieur à $L2 = [x, y]$" si :

$$a < x \quad \text{ou} \quad (a = x \text{ et } b \leqslant y).$$

Écrire une fonction `less` de deux listes de longueur deux `L1` et `L2` qui renvoie `True` si "$L1$ est inférieur à $L2$" et `False` sinon.

4 * En s'inspirant du code définissant la fonction `ordonner`, écrire une fonction `ordonnerDansR2` qui utilise `less` et qui ordonne une liste de couples $[x, y]$ selon la relation d'ordre définie précédemment.

5 * Créer une liste `A` de flottants allant de $-\pi$ à π, avec un pas de $\pi/100$.
Le nombre π est dans la bibliothèque `math`, sous le nom `pi`.

6 Créer la liste `L` des listes $[(20 + \cos(10a))\cos(a), (20 + \cos(10a))\sin(a)]$, pour a parcourant la liste `A`.
Les fonctions cos et sin sont dans la bibliothèque `math`.

7 Ordonner les points de `L` avec `ordonnerDansR2`.

8 Faire tracer dans le plan le nuage de points ainsi que la ligne les reliant. On utilisera `plot` et `show` du module `matplotlib.pyplot`.

Exercice 4

Dans une chaîne de caractères, on appelle "composante" une sous-chaîne correspondant à la répétition d'un même caractère, précédée et suivie par rien ou un autre caractère.
Les composantes de 'aaaaggbbbzdaa' sont : 'aaaa' suivie de, 'gg', 'bbb', 'z', 'd' et 'aa'.

1 * Écrire une fonction `comptage` qui prend comme argument une chaîne de caractères S et renvoie un dictionnaire dont chaque clé est un caractère différent de S et l'élément associé à la clé est le nombre de fois que ce caractère apparaît dans S.
Par exemple, `comptage('aaaaggbbbzdaaa')` renvoie `{'a':7,'g':2,'b':3,'z':1,'d':1}`.
Afficher le résultat `comptage('aaaaggbbbzdaaa')`.

2 * Écrire une fonction `nbelem` dont l'argument est une chaîne de caractères S et qui renvoie le nombre de caractères distincts contenus dans S. Par exemple, `nbelem('aaaaggbbbzdaaa')` donne 5.
Afficher le résultat `nbelem('aaaaggbbbzdaaa')` .

3 * Écrire une fonction `nbits` dont l'argument est un entier naturel $n \in \mathbb{N}^*$ et qui renvoie le plus petit entier $p\mathbb{N}^*$ tel que $n \leqslant 2^p$.
Afficher le résultat `nbits(5)`.

4 Sur l'exemple 'aaaaggbbbzdaaa', on a donc 5 lettres à coder. Il suffit pour ça de 3 bits avec le code : 'a' : 000,'b' : 001, 'd' :010, 'g' : 011, 'z' : 100.
Écrire une fonction `nbtotalbits` dont l'argument est une chaine de caractère S et qui renvoie le nombre minimum de bits nécessaires pour coder les lettres de S.
Afficher le résultat `nbtotalbits('aaaaggbbbzdaaa')`.

5 Écrire une fonction `coupures` dont l'argument est une chaîne de caractères S, qui renvoie la liste des indices i tels que $S[i] \neq S[i-1]$. Par exemple, coupures('aaaaggbbbzdaaa') donne $[4, 6, 9, 10, 11]$. Afficher le résultat coupures('aaaaggbbbzdaaa').

6 Si S est une liste composée de 45 'a' suivis de 71 'b' , quelle solution peut-on trouver pour minimiser le stockage de S ? Réponse en commentaire.