NSI : TP n°3, vers la notion de tuples en Python.

Conjecture de Syracuse, ou de Collatz, ou d'Ulam….

> “Les mathématiques ne sont pas encore prêtes pour de tels problèmes”. Paul Erdos

Soit N un entier naturel, on applique les instructions suivantes :

- si N est pair , on le divise par 2,
- si N est impair, on le multiplie par 3 et on ajoute 1.
- On applique à nouveau au nombre obtenu cet algorithme.

1. a) Appliquer ces instructions au nombre $N = 14$. Que remarquez vous ?
   b) Appliquer ces instructions à un entier de votre choix. Que remarquez-vous ?
2. Peut-on définir cette série d'instructions comme un algorithme ? Que faut-il modifier à cette série d'instructions pour constituer un algorithme ?
3. Compléter la fonction python suivante qui retourne ce que l'on appelle le vol du nombre N, c'est à dire la suite des nombres obtenus en appliquant cet algorithme jusqu'à obtenir 1.

```
def vol(n):
  ''' retourne le vol du nombre n en appliquant l'algorithme de
syracuse
  : return : une chaîne de caractères (str)

  >>> vol(1)
  '1,4,2,1'
  >>> vol(5)
  '5,16,8,4,2,1'
  '''
```

4. On désire à présent modifier la fonction vol pour qu'elle retourne un couple de valeurs : le vol du nombre n et la longueur de ce vol, c'est à dire le nombre d'étapes nécessaire pour atterir à la valeur 1.

```
def vol(n):
  ''' retourne le vol de n et la longueur de ce vol
  : return : un tuple : vol,longueur_vol
  vol : str
  longueur_vol : int

  >>> longueur_vol(1)
   '1,4,2,1', 4
  >>> longueur_vol(5)
  '5,16,8,4,2,1', 6
  '''
```

5. Modifier à présent la fonction vol pour qu'elle retourne un triplet de valeurs :
   - le vol du nombre n,
   - la longueur du vol du nombre n,
   - la hauteur maximale atteinte durant le vol du nombre n.

```
def vol(n):
  ''' retourne le vol de n , la longueur de ce vol et la hauteur
maximale
  : return : un tuple : vol,longueur_vol,hauteur_max
  vol : str
  longueur_vol : int
  hauteur_max : int

  >>> longueur_vol(1)
   '1,4,2,1',4,4
  >>> longueur_vol(5)
  '5,16,8,4,2,1',6,16
  '''
```

6. a) Ajouter une fonction vol_longueur_max(n) qui détermine pour quelle valeur de k , comprise entre 1 et n, on a le vol le plus long.
   b) En déduire quel est l'entier n compris entre 1 et 100 inclus qui a le vol le plus long. Et pour n entre 1 et 1 000 ?
7. a) Ajouter une fonction vol_hauteur_max(n) qui détermine pour quelle valeur de k, comprise entre 1 et n, on a la hauteur de vol la plus grande.
   b) En déduire quel est l'entier n compris entre 1 et 100 inclus qui a le vol le plus haut. Et pour n entre 1 et 1 000.

> "Une course au record est engagée de la plus grande valeur vérifiant la conjecture. Le record de 2009 tenu par T. Oliveira e Silva est de $5 \times 2^{60} \approx 5.8 \times 10^{18}$(Oliveira, 2010, [46])."
>
> Luc-Olivier Pochon, Alain Favre : https://hal.archives-ouvertes.fr/hal-01593181/document

Remarque : pour la question 4, on pourrait penser à utiliser la méthode count des chaînes de caractères :

```
>>> foin = "Le héron au long bec emmanché d'un long cou"
>>> aiguille = 'long'
>>> print foin.count(aiguille)
2
```

https://fr.wikibooks.org/wiki/Programmation_Python/Chaines_de_caract%C3%A8res

TP inspiré de : https://fr.wikipedia.org/wiki/Conjecture_de_Syracuse