

Petits exercices en python

1 Range

Exercice 1. Que renvoient les commandes suivantes (répondre sur une feuille de papier, puis vérifier à l'aide de l'interpréteur python) :

1. `list(range(5))`
2. `list(range(2, 8, 2))`
3. `list(range(6, 1))`
4. `list(range(10, 0, -2))`

2 Petits exercices avec des boucles for

Exercice 2. Ecrire une fonction d'entête `def somme(n)` : qui admet en argument un entier naturel `n` et qui renvoie la somme suivante :

$$\sum_{k=0}^n \frac{k}{k^2 + 1}$$

Exercice 3. Même question qu'à l'exercice précédent pour calculer plutôt la somme :

$$\sum_{k=0}^n \sum_{j=k}^n \frac{k}{j^2 + 1}$$

Exercice 4. Maxima locaux d'une liste. Etant donnée une liste `L` de nombres, on cherche à extraire la sous-liste formée des maxima locaux de `L`. On dira d'un terme `L[i]` de `L` qu'il est un maximum local si et seulement si une des conditions qui suivent est réalisée :

- si `i=0` et `L` ne comporte qu'un élément
- si `i=0` et `L[0] >= L[1]`
- si `0 < i < len(L) - 1` et `L[i-1] <= L[i] >= L[i+1]`
- si `i=len(L)-1` et `L[i-1] <= L[i]`

On associera ainsi à la liste `[5, 2, 3, 3, 2, 6, 1]` la liste `[5, 3, 3, 6]`.

Ecrire une fonction d'entête `def maximaLocaux(L)` : qui admet en argument une liste supposée non vide `L` et qui renvoie la liste extraite de `L` formée de ses maxima locaux.

Indication : pour construire cette liste extraite, on partira d'une liste vide, et on utilisera la méthode `append` pour y incorporer au fur et à mesure tous les maxima locaux trouvés.

3 Boucles while

Exercice 5. La célèbre conjecture de Syracuse est la suivante : on part d'un entier strictement positif `n`, et on lui associe ou bien `3n + 1` si `n` est impair, ou bien `n/2` si `n` est pair, et on recommence, jusqu'à parvenir à l'entier 1 (quite à ce qu'on boucle infiniment bien sûr). La conjecture de Syracuse est qu'on parvient toujours à l'entier 1. Cette conjecture, quoiqu'en pensent certains, reste à ce jour un problème ouvert.

Pour expérimenter un peu, écrire une fonction d'entête `def syracuse(n)` : qui prend un argument un entier strictement positif, et qui renvoie le nombre d'étapes en partant de `n` qui conduisent à l'entier 1.

Voici les premières valeurs :

<code>n</code>	1	2	3	4	5
<code>syracuse(n)</code>	0	1	7	2	5

Exercice 6. A l'aide de la fonction écrite à la fonction précédente, déterminer le plus petit entier `n` tel que `syracuse(n) > 300`.