TP # 17

Conjecture de Syracuse
(Représentation de suites)

Rappel sur les représentations de suites

Représenter les premières valeurs d'une suite $(u_n)_{n\in\mathbb{N}}$ consiste à relier les points du plan :

$$M_0 = (0, u_0),\ M_1 = (1, u_2),\ M_2 = (2, u_2)\ \ldots,\ M_n = (n, u_n).$$

Il faut donc :

- Créer le vecteur des abscisses : `X = [` $0, 1, \ldots, n$ `]` (avec, ici, `X = np.arange(n+1)`)
- Construire le vecteur des ordonnées : `Y = [` $u_0, u_1, \ldots, u_n$ `]`
(directement pour une suite explicite, avec une boucle pour une suite récurrente...)
- Relier les points avec `plt.plot(X,Y)` puis afficher la représentation avec `plt.show()`.

La **suite de Syracuse** issue de $a \in \mathbb{N}^*$ (la "graine") est définie par :

$$u_0 = a \ \text{ et } \ \forall n \in \mathbb{N},\ u_{n+1} = \begin{cases} \frac{u_n}{2} \text{ si } u_n \text{ est pair} \\ 3u_n + 1 \text{ sinon.} \end{cases}$$

Questions préliminaires :

1. Calculer les premiers termes de la suite de Syracuse issue de 1.

2. Calculer les premiers termes de la suite de Syracuse issue de 5.

3. Que dire, donc, s'il existe un rang $n \in \mathbb{N}$ tel que $u_n = 1$?

La **Conjecture de Syracuse** (non démontrée à ce jour) s'énonce ainsi :

"Quel que soit $a \in \mathbb{N}^*$, la suite de Syracuse issue de a atteint la valeur 1."

Toutes les fonctions demandées dans ce TP sont à écrire à la suite les unes des autres, dans l'éditeur de texte.

1. **Disjonction.**
Définir en Python une fonction `f` qui prend en entrée un entier x et renvoie :
$x/2$ si x est pair, $3x + 1$ sinon. (On se souviendra de l'opération `%` en Python !)

2. **Représentation de la suite de Syracuse.**
Compléter le programme pour que `syracuse(a,n)` affiche la représentation graphique des termes $u_0, u_1, \ldots, u_n$ de la suite de Syracuse issue de a.

```
import numpy as np
import matplotlib.pyplot as plt

def syracuse(a,n) :

        X = .................

        Y = ................. ; Y[0] = .......

        for k in range(.........) :

                Y[k+1] = ............


        ...............................


        ...............................
```

Représenter ainsi les premiers termes des suites de Syracuse issues de :

$$a = 1,\ \ 5,\ \ 13,\ \ 19,\ \ 27,\ \ 121.$$

On adaptera le nombre de termes affichés de sorte à pouvoir confirmer visuellement la conjecture de Syracuse dans ces différents cas !

3. **Durée du vol.**
Compléter le programme pour que `indice(a)` renvoie le plus petit indice $n \in \mathbb{N}$ tel que $u_n = 1$ (où u est toujours la suite de Syracuse issue de a).

On note ce "plus petit indice" N_a, et on l'appelle la "durée de vol".

```
def indice(a) :

        n = ........ ;    u = ..........

        while ...................... :

                n = n + 1
                u = f(u)

        return( ..... )
```

Donner ainsi les "durées de vol" pour les valeurs de a suivantes :

$N_1 = ...$ $N_5 = ...$ $N_{13} = ...$ $N_{27} = ...$ $N_{121} = ...$ $N_{999} = ...$

4. **"Durée de vol" en fonction de a**
Donner les instructions permettant de représenter graphiquement
les 1000 premières durées de vol : $N_1, N_2, N_3, \ldots, N_{1000}$.

```

```

Pour plus de lisibilité, **on pourra afficher uniquement des points (sans les "traits")** à l'aide de l'option `'.'` dans `plt.plot`.

Cette représentation permet de confirmer la conjecture de Syracuse pour $a \in [\![1, 1000]\!]$!

S'il vous reste du temps, on pourra tenter de confirmer la conjecture jusqu'à $a = 10000$, $a = 100000$... en tachant de ne pas faire cramer son ordinateur.