1G. Correction du devoir maison n°9.
La suite de Syracuse

Exercice 1

On considère les suites (u_n) définies de la façon suivante :
Le premier terme u_0 est un nombre entier positif donné. Pour tout $n \geqslant 0$,

— si u_n est pair alors $u_{n+1} = \dfrac{u_n}{2}$

— si u_n est impair alors $u_{n+1} = 3u_n + 1$

Première partie

1. Calculer les 9 premiers termes de la suite si $u_0 = 5$.

n	0	1	2	3	4	5	6	7	8
u_n	5	16	8	4	2	1	4	2	1

2. Soit (u_n) une des suites définies précédemment. Supposons qu'il existe un entier n_0 tel que $u_{n_0} = 1$. Que peut-on dire des termes de la suite à partir de n_0 ?

À partir du rang n_0, la suite devient périodique, elle répète le motif 4 ;2 ;1 à l'infini.

3. Compléter la fonction `syracuse` d'arguments u et n qui renvoie la liste des $n+1$ premiers termes de la suite pour une valeur de u_0 saisie dans la variable u.

```
def syracuse(u,n):
    L=[u]
    for i in range(1,n+1):
        if u%2==0 :
            u=u/2
        else :
            u=3*u+1
        L=L+[u]
    return(L)
```

4. Finit-on par obtenir 1 si le premier terme de la suite est $u_0 = 3$? $u_0 = 7$? $u_0 = 11$? $u_0 = 13$? $u_0 = 19$?

Oui à chaque fois, pour ces valeurs de u_0, il est suffisant de tester la fonction précédente avec $n = 20$ pour le vérifier.

Deuxième partie

1. **Temps de vol**

On conjecture que quel que soit l'entier positif u_0, u_n atteint la valeur 1.
Cette conjecture n'est à ce jour pas démontrée, et on n'a pas trouvé de contre-exemple.
On appelle temps de vol de la valeur k, le plus petit entier n tel que $u_n = 1$, obtenu en prenant $u_0 = k$.

(a) Déterminer le temps de vol de la suite du 1 ($u_0 = 5$).

D'après la question 1 de la partie 1, pour $u_0 = 5$, le plus petit entier n tel que $u_n = 1$ est 5, donc le temps de vol est 5.

(b) Compléter la fonction Python qui renvoie le temps de vol pour une valeur de u_0 saisie en argument.

```
def tempsdevol(u):
    i=0
    while u!=1 :
        if u%2==0:
            u=u/2
        else :
            u=3*u+1
        i=i+1
    return(i)
```

(c) Donner les temps de vol des suites du 4.

Valeur de u_0	3	7	11	13	19
Temps de vol	7	16	14	9	20

2. **Altitude**

On appelle altitude de la valeur k, la plus grande valeur de u_n obtenue en prenant $u_0 = k$.

(a) Donner l'altitude de la suite du 1. ($u_0 = 5$).

L'attitude est 16 pour $u_0 = 5$.

(b) Écrire en Python une fonction `altitude` qui renvoie l'altitude lorsqu'on entre la valeur de u_0 en argument.

```
def altitude(u):
    n=tempsdevol(u)
    return(max(syracuse(u,n)))
```

(c) Donner à l'aide de cette fonction les altitudes des suites du 4.

Valeur de u_0	3	7	11	13	19
Altitude	16	52	52	40	88