

En mathématiques, on appelle suite de Syracuse une suite d'entiers naturels définie de la manière suivante : on part d'un nombre entier plus grand que zéro ; s'il est pair, on le divise par 2 ; s'il est impair, on le multiplie par 3 et on ajoute 1. En répétant l'opération, on obtient une suite d'entiers positifs dont chacun ne dépend que de son prédécesseur.

Ces opérations permettent de construire la suite u_n telle que , $u_0 \in \mathbb{N}^*$ puis,

$$\forall n \in \mathbb{N}^*, u_{n+1} = \frac{u_n}{2} \text{ si } u_n \text{ est pair et } u_{n+1} = 3u_n + 1 \text{ si } u_n \text{ est impair}$$

Ainsi à partir de $u_0 = 5$ apparait la suite de nombres : 5,16,8,4,2,1 puis à partir de 1, à nouveau 4,2,1 et ainsi de suite pour un cycle sans fin.

La conjecture de Syracuse, est l'hypothèse mathématique selon laquelle la suite de Syracuse de n'importe quel entier strictement positif atteint 1. En dépit de la simplicité de son énoncé, cette conjecture n'a toujours pas trouvé de preuve malgré de nombreuses tentatives de mathématiciens.

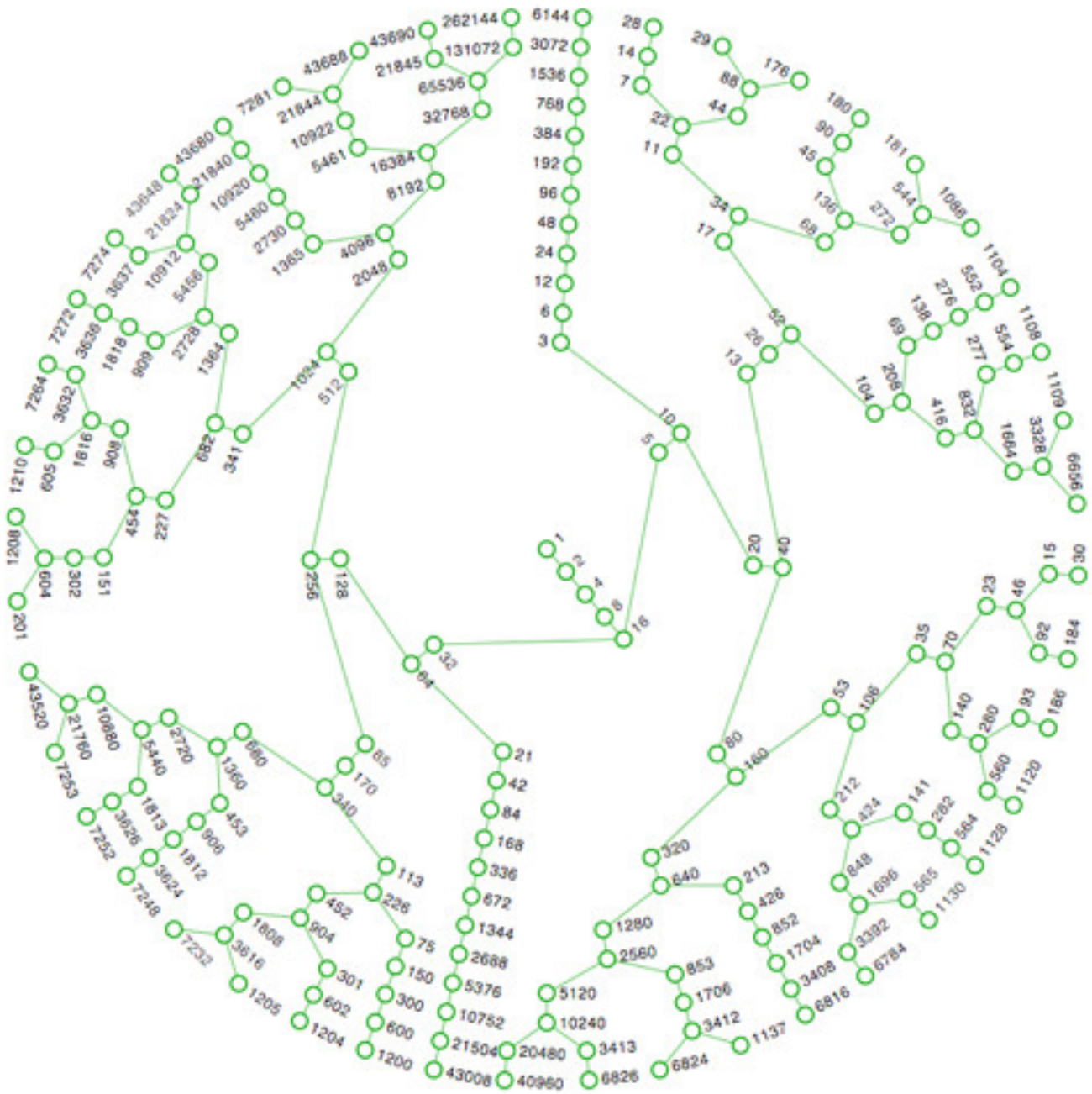


Figure 1 – Les suites de Syracuse à temps de vols allant de 1 à 18

Dans le cas de la suite de Syracuse, le **temps de vol** correspond au nombre de valeurs prises par la suite avant de retomber à 1 et l'**altitude** correspond à la valeur maximale atteinte par un terme de la suite.

1 Algorithme de la suite de Syracuse

1. Définir une fonction **syracuse**(u_0, n) qui prend en entrée une valeur d'initialisation u_0 et un indice n et renvoie la valeur de u_n
2. Définir une fonction **syracuseListe**(u_0, n) qui prend en entrée une valeur d'initialisation u_0 et un indice n et renvoie l'ensemble des valeurs de u_k pour $k \in [1, n]$ sous forme d'une liste de valeurs.
3. Tracer u_n en fonction de n pour une suite de Syracuse partant d'une des valeurs présentes sur l'illustration.
4. Vérifier sur plusieurs exemples la conjecture de Syracuse.

2 Caractérisation d'une suite de Syracuse : Altitude et temps de vol

5. Définir une fonction **maxi**(**listeNumerique**) qui prend en entrée une liste de valeurs et renvoie un tuple contenant la valeur du maximum de la liste et sa position dans la liste.
6. Définir une fonction **vol**(**suite**) qui prend en entrée une suite de Syracuse et renvoie son temps de vol
7. Modifier la fonction **syracuseListe**(u_0, n) afin que :
 - la fonction n'ait plus qu'un seul argument en entrée correspondant à u_0 ;
 - la liste se termine une fois atteinte la première occurrence de la valeur 1 ;
 - la fonction renvoie la liste de valeur, l'altitude et sa position, le temps de vol.
8. Que doit renvoyer la commande **syracuseListe**(181) ?
9. Quelle est le type chaque valeur de sortie ?
10. Tester la fonction pour vérifier qu'elle renvoie bien les valeurs attendues

3 Étude graphique des suites de Syracuse à long temps de vol

11. Définir une fonction **altitudes**(u_0max) qui construit une liste des altitudes des suites de Syracuse initialisées par $u_0 \in [1; u_0max]$.
12. Définir une fonction **vols**(u_0max) qui construit une liste des temps de vols des suites de Syracuse initialisées par $u_0 \in [1; u_0max]$.
13. Tracer sur deux graphes séparés les altitudes et temps de vols pour les suites de Syracuse avec des valeurs de départ comprises entre 1 et 1000