IUT Saint-Nazaire UNIVERSITÉ DE NANTES

EN LETTRES CAPITALES NOM(S) :	PRÉNOM(S) :	GROUPE :

– Travaux pratiques de Mathématiques –

LA CONJECTURE DE SYRACUSE

La conjecture de Syracuse, encore appelée conjecture de Collatz, conjecture d'Ulam, conjecture tchèque ou problème « 3x+1 » 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 défie depuis de nombreuses années les mathématiciens. Paul Erdös a dit à propos de la conjecture de Syracuse : « les mathématiques ne sont pas encore prêtes pour de tels problèmes ».

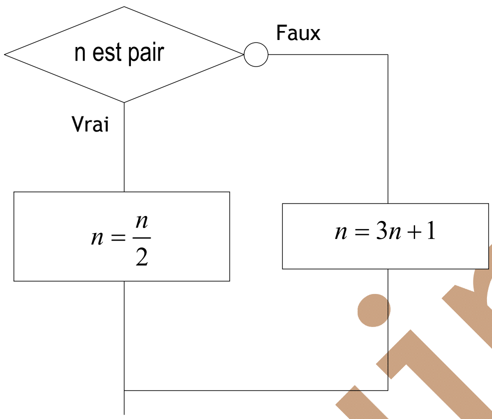
1- La suite de Syracuse

1-1. Enoncé

On considère une suite d'entiers naturels définie de la manière suivante :
On part d'un nombre entier donné plus grand que zéro :

- si il est pair, on le divise par 2 ;
- si 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 des entiers ne dépend que de son prédécesseur.



Par exemple, à partir de 14, on construit la suite d'entiers naturels :
14, 7, 22, 11, 34, 17, 52, 26, 13, 40, 20, 10, 5, 16, 8, 4, 2, 1, 4, 2…
Il s'agit de la suite d'entiers relative au nombre 14.

Dès que le nombre 1 a été atteint, la suite des valeurs 1, 4, 2, 1, 4, 2 … se répète indéfiniment en un cycle de longueur 3, appelé cycle trivial.

Si l'on était parti d'un autre entier, en lui appliquant les mêmes règles, on aurait obtenu une suite de nombres différente. A priori, il serait possible que cette suite, à partir de certaines valeurs de départ n'atteigne jamais la valeur 1, soit qu'elle aboutisse à un cycle différent du cycle trivial, soit qu'elle diverge vers l'infini. Or, on n'a jamais trouvé d'exemple de suite obtenue suivant les règles données qui n'aboutisse à 1 et, par suite, au cycle trivial.

1-2. Travail demandé

1] Ecrire l'algorithme permettant de calculer les 100 premières valeurs de cette suite pour un nombre n donné.
2] Ecrire le programme correspondant permettant de saisir l'entier n.
3] On appelle la durée de vol le nombre d'entiers qui précède la première apparition du nombre 1. Par exemple, l'entier 14 a une durée de vol égale à 17. Modifier le programme afin qu'il affiche la durée de vol de l'entier donné. Utiliser ce programme et donner les durées de vol des entiers A = 27, B = 1245 et C = 1537.
4] Modifier le programme afin qu'il recherche l'entier N qui a la plus grande durée de vol sachant que N est compris entre a et b donnés par l'utilisateur. Déterminer l'entier N compris entre 145 et 345.

Remarque générale au niveau de la présentation de vos résultats dans la console Scilab :
Soignez la mise en forme de votre interface utilisateur :

- Ecrivez « Valeur de n = » afin que l'utilisateur sache quelle valeur il doit saisir ;
- Ecrivez « Résultats x = etc. » afin que l'utilisateur sache de quel résultat il s'agit

Bref, il faut présenter vos résultats et pas seulement des valeurs numériques brutes.

1-3. Algorithmique

1-3.1 Les données

n : L'entier à partir duquel on calcule la suite d'entiers naturels.

1-3.2 Les valeurs à déterminer

Les 100 premières valeurs de la suite d'entiers.

1-3.3 La méthode utilisée

- On teste la parité du nombre.
- On effectue une des deux opérations suivant le résultat du test.

1-3.4 Les fonctions et structures

➔ Structure répétitive
Pour k **de** 1 **jusqu'à** n **Faire**
{Traitement 1}
FinPour

➔ Structure alternative
Si {condition} **Alors**
{Traitement 1}
Sinon
{Traitement 2}
FinSi

➔ modulo(a , m)
Renvoie le reste de la division euclidienne de a par m.
Exemple :
x = modulo(23 , 4) ; x contient la valeur 3 car 23 = 4*5 + 3
y = modulo(5 , 2) ; y contient la valeur 1
z = modulo(8 , 2) ; z contient la valeur 0

➔ t = zeros(1 , 100)
La fonction zeros(n , p) définit une matrice de n lignes et de p colonnes dont tous les termes sont nuls.
La fonction zeros(1 , 100) définit un vecteur ligne de 100 colonnes dont tous les termes sont nuls.
t(1 , 2) = 6 place la valeur 6 dans la deuxième colonne du vecteur ligne t.

➔ disp(t)
disp(t) : Affiche les éléments du vecteur ligne t.

➔ Affichage de plusieurs variables
```
printf ("Encadrement : %f%s%f\n",a," < xsol < ",b);
```
L'affichage est le suivant : Encadrement : 3.412 < xsol < 3.413

➔ break
L'instruction break permet d'interrompre l'exécution d'une boucle for

➔ return
L'instruction return permet de sortir d'une fonction et de retourner à l'endroit de l'appel.