

TP 4 : Structure itérative WHILE

Exercice 1 Rappel et découverte de la commande Scilab

1. On considère l'algorithme suivant, écrit en langage naturel :

```
x prend la valeur 2
Tant que x < 100
    x prend la valeur 2 x + 1
FinPour
Afficher x
```

Sans utiliser l'ordinateur, exécutez l'algorithme en complétant le tableau suivant :

Etape	1	2	3	4	5	6	7	8
Valeur de S								

Quelle valeur est affichée ?

2. Expliquez le lien entre cet algorithme et la suite définie par $u_0=2$ et $u_{n+1}=2u_n+1$.

3. Ouvrez Scilab, puis entrez et exécutez le script suivant :

```
1 x=2
2
3 while x<100
4     x=2*x+1
5 end
6
7 disp(x)
```

a) Vérifiez la réponse à la question précédente.

b) Trouvez le rang n du premier terme u_n qui dépasse 1000.

c) Est-ce que la suite (u_n) possède un terme supérieur ou égal à un million ?

Exercice 2 Structure for construite à partir d'une structure while

1. Reproduisez le script suivant dans Scinotes, et exécutez-le à la main pour vérifier le résultat affiché.

```
1 x=2
2 i=0
3
4 while i<=4
5     x=x^2+1
6     i=i+1
7     disp(x,i)
8 end
```

Etape	1	2	3	4	5
Valeur de x					
Valeur de k					

2. Expliquez ce que calcule ce programme.

Exercice 1 Algorithme de seuil

On considère l'algorithme suivant :

```
Saisir M
N prend la valeur 0
Tant que  $3^N < M$ 
    N prend la valeur N+1
Fin Tant que
Afficher N
```

1. Créez l'algorithme dans Scilab et l'exécuter avec $M=20$, $M=100$ et $M=10^6$.

1	
2	
3	
4	
5	
6	
7	
8	

2. Que fait cet algorithme ?

Exercice 2 Calcul des termes d'une suite

1. Sans utiliser de boucle for, créer un programme qui calcule et qui affiche les termes de la suite (u_n) définie par

$$u_0=6 \text{ et } u_{n+1}=2u_n-5 \text{ jusqu'à l'indice } n=15.$$

2. Créez un programme qui calcule et affiche les termes de cette suite jusqu'à ce que $u_n \geq 100$

1	
2	
3	
4	
5	
6	
7	
8	

1	
2	
3	
4	
5	
6	
7	
8	

Exercice 3 Calcul d'un placement théorique

On place une somme de 20€ à la banque avec un taux d'intérêt annuel fixe de 2,5%.

1. Écrire un algorithme qui calcule la somme dont on disposera dans 20 ans.
2. Écrire un algorithme qui calcule le temps qu'il faut pour être millionnaire.

1	
2	
3	
4	
5	
6	
7	
8	

1	
2	
3	
4	
5	
6	
7	
8	