

Algorithmes (6) Boucles « Répéter »

Objectif : étudier une nouvelle structure répétitive.

I. Exemple d'algorithme calculant des augmentations successives de 4 %

1°) Situation étudiée

On cherche un algorithme donnant le nombre d'augmentations successives de 4 % nécessaires pour dépasser 6000, en partant de 5000.

2°) Analyse du problème

On va utiliser une structure de boucle avec test d'arrêt ; on a deux structures possibles : boucle « Tantque » ou boucle « Répéter ».

3°) Algorithme rédigé en langage naturel

Initialisations :
S prend la valeur 5000
n prend la valeur 0

Traitement :

Répéter

S prend la valeur $1,04 \times S$
n prend la valeur $n + 1$

Jusqu'à $S > 6000$

Sortie :

Afficher n

II. Syntaxe en langage naturel

Répéter
suite d'instructions
Jusqu'à *condition*

Après avoir exécuté la suite d'instructions, on teste si la condition est vraie.
Si elle est vraie, on s'arrête et on sort de la boucle.
Si elle est fausse, on recommence la suite d'instructions.

On peut toujours remplacer une boucle « Répéter » par une boucle « Tantque ».
C'est pourquoi certains logiciels ne sont pas équipés de cette boucle.

III. Programmation sur calculatrice

Sur les calculatrices, il s'agit d'une structure « Répète... Tant que » et non « Répète... Jusqu'à ».
Donc il faut faire attention lors de l'écriture de la condition (testée en fin de boucle).

Calculatrice TI	Calculatrice Casio
: 5000 → S	5000 → S ↓
: 0 → N	0 → N ↓
: Repeat S > 6000	Do ↓
: S * 1.04 → S	S * 1.04 → S ↓
: N + 1 → N	N + 1 → N ↓
: End	LpWhile S ≤ 6000 ↓
: Disp N	"N=" : N ▲

On trouve 5 répétitions.

On dépassera 6000 euros au bout de 5 années.

IV. Comparaison des structures « Tantque » et « Répéter »

On peut utiliser une boucle « Répéter » dans les mêmes conditions qu'une boucle « Tantque » (nombre d'itérations non connu à l'avance ».

Après une structure « Répéter jusqu'à », la suite d'instructions est exécutée au moins une fois ; tandis qu'avec « Tant que », elle peut ne pas l'être puisqu'on peut ne pas entrer dans la boucle.

Appendice : bilan sur les structures répétitives

- boucle pour (nombre d'itérations connu à l'avance)
- boucle « Tantque »
- boucles « Répéter jusqu'à »

Exercices

1 On considère l'algorithme ci-dessous.

```
Initialisation :  
S prend la valeur 1  
  
Traitement et sorties :  
Répéter  
| Afficher S  
| S prend la valeur S + 1  
Jusqu'à S ≥ 5
```

Recopier cet algorithme.

- 1°) Faire fonctionner l'algorithme à la main.
Indiquer quels sont les affichages que l'on obtient.
- 2°) Programmer cet algorithme sur calculatrice ou sur ordinateur et vérifier le résultat de la question précédente.
- 3°) Écrire un algorithme avec une boucle « Tantque » qui permette d'obtenir les mêmes affichages.

2 On considère l'algorithme ci-dessous :

```
Initialisation :  
u prend la valeur 2  
  
Traitement et sorties :  
Tantque u > 1,001 Faire  
| Afficher u  
| u prend la valeur  $\sqrt{u}$   
FinTantque
```

Recopier cet algorithme.

1°) Recopier et compléter la phrase suivante :

L'algorithme affiche tous les termes de la suite (u_n) définie sur \mathbb{N} par son premier terme $u_0 = \dots$ et par la relation de récurrence $\forall n \in \mathbb{N} \quad u_{n+1} = \dots$ tant que

2°) a) Recopier et compléter la phrase suivante :

L'algorithme affiche tous les termes de la suite (u_n) jusqu'à ce que

b) Réécrire l'algorithme à l'aide de la structure « Répéter ... jusqu'à ... ».

3 On considère l'algorithme ci-dessous.

```
Initialisations :  
S prend la valeur 0  
i prend la valeur 0  
  
Traitement :  
Répéter  
| i prend la valeur i + 1  
| S prend la valeur S +  $\sqrt{i}$   
Jusqu'à S > 10  
  
Sortie :  
Afficher S
```

Recopier cet algorithme.

- 1°) Que fait cet algorithme ?
- 2°) Réécrire cet algorithme à l'aide d'une boucle « Tantque ».
- 3°) Programmer l'algorithme dans la version initiale et indiquer le résultat obtenu en sortie.

Corrigé

1

Initialisation :
S prend la valeur 1

Traitement et sorties :

Répéter
| Afficher S
| S prend la valeur S + 1
Jusqu'à S \geq 5

1°) Faire fonctionner l'algorithme à la main.
Indiquer quels sont les affichages que l'on obtient.

On obtient mes affichages : 1 ; 2 ; 3 ; 4 ; 5.

Afficher 1
S prend la valeur 2

Afficher 2
S prend a valeur 3

Afficher 3
S prend la valeur 4

Afficher 4
S prend la valeur 5

On s'arrête là.

La condition est vraie. On arrête de faire tourner l'algorithme.

2°) Programme sur calculatrice

Calculatrice TI

```
: 1 → S  
: Repeat S  $\geq$  5  
: Disp S  
: S + 1 → S  
: End
```

3°) Écrire un algorithme avec une boucle « Tantque » qui permette d'obtenir les mêmes affichages.

Initialisation :
S prend la valeur 1

Traitement et sorties :

Tantque S < 5 **Faire**
| Afficher S
| S prend la valeur S + 1
FinTantque