# Notion d’exécution répétée (ou boucles)

Dans une procédure, il arrive parfois que des opérations doivent être répétées plusieurs fois.

Dans la vie courante, on est très souvent confronté à cette situation :

Par exemple, si on doit remplir une baignoire avec des seaux (oui, ça peut arriver).

* **tant que** la baignoire n’est pas pleine **répéter** verser un seau.
	+ la condition, exprimée sous forme d’une expression logique, est : «la baignoire n’est pas pleine»
	+ si cette condition est vraie, et tant qu’elle est vraie, on va répéter «verser un seau»
	+ dans le cas contraire, on ne fait rien

# Structure itérative

Initialisation des valeurs d'une condition

**TANTQUE** (condition)

 **REPETER**

liste des opérations à répéter

modification des valeurs de la condition

**finTANTQUE**

La condition est exprimée de la même manière que dans la fiche02 relative aux structures conditionnelles.

# Exemple : multiplier les nombres de 1 à 4 par 12 🡪 réflexion, démarche

* calculer 1x12, puis 2x12, puis 3x12, et enfin 4x12
* soit, si on donne le nom *compteur* au nombre qui varie de 1 à 6 :
	+ donner la valeur 1 à compteur
	+ tant que compteur est inférieur ou égal à 4,
		- calculer *compteur* x 12
		- ajouter 1 à *compteur*

# Exemple : multiplier les nombres de 1 à 4 par 12 🡪 procédure

L’algorithme suivant affiche la table de multiplication des nombres de 1 à 4 par 12.

DECLARATIONS

CONSTANTE

 MULT : entier 🡨 12

VARIABLES

 compteur : entier 🡨 0 // compteur de répétition

 resultat : entier 🡨 0 // resultat de la multiplication

DEBUT (des opérations)

// afficher la table de multiplication

compteur 🡨 1 🡸 initialisation

 **TANTQUE** (compteur <= 4) 🡸 test

 **REPETER**

 // calcul et affichage du résultat

 resultat 🡨 compteur \* MULT

 AFFICHER compteur, " X ", MULT, " = ",resultat

 // ajouter 1 au compteur

 compteur 🡨 compteur + 1 🡸 modification

**finTANTQUE**

AFFICHER « terminé ! »

FIN (des opérations)

## Trace de l’exécution

La trace d'exécution est un outil qui permet de vérifier le bon fonctionnement d'une procédure, surtout dans le cas des itérations. Les différentes opérations de l'algorithme sont numérotées et on trace son exécution :

DECLARATIONS

CONSTANTE

 MULT : entier 🡨 12

VARIABLES

 compteur : entier 🡨 0 // compteur de répétition

 resultat : entier 🡨 0 // resultat de la multiplication

DEBUT (des opérations)

// afficher la table de multiplication

01 compteur 🡨 1 🡸 initialisation

02 **TANTQUE** (compteur <= 4) 🡸 test

 **REPETER**

 // calcul et affichage du résultat

03 resultat 🡨 compteur \* MULT

03bis AFFICHER compteur, " X ", MULT, " = ",resultat

 // ajouter 1 au compteur

04 compteur 🡨 compteur + 1 🡸 modification

 **finTANTQUE**

05 AFFICHER « terminé ! »

FIN (des opérations)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Pas** | **Entrée** | **Condition****Vrai / Faux** | **CONSTANTES VARIABLES** | **Sortie** |
|  | **MULT** |  |  | **compteur** | **resultat** |  |  |
| **DEB** |  |  |  | 12 |  |  | 0 | 0 |  |  |  |
| 01 |  |  |  |  |  |  | 1 |  |  |  |  |
| 02 |  | compteur<= 4 ? Vrai |  |  |  |  |  |  |  |  |  |
| 03 |  |  |  |  |  |  |  | 1\*12 =12 |  |  | 1x12-12 |
| 04 |  |  |  |  |  |  | 1+1 = 2 |  |  |  |  |
| 02 |  | compteur<= 4 ? Vrai |  |  |  |  |  |  |  |  |  |
| 03 |  |  |  |  |  |  |  | 2\*12 = 24 |  |  | 2x12=24 |
| 04 |  |  |  |  |  |  | 2+1 = 3 |  |  |  |  |
| 02 |  | compteur<= 4 ? Vrai |  |  |  |  |  |  |  |  |  |
| 03 |  |  |  |  |  |  |  | 3\*12=36 |  |  | 3x12=36 |
| 04 |  |  |  |  |  |  | 3+1 = 4 |  |  |  |  |
| 02 |  | compteur<= 4 ? Vrai |  |  |  |  |  |  |  |  |  |
| 03 |  |  |  |  |  |  |  | 4\*12=48 |  |  | 4x12=48 |
| 04 |  |  |  |  |  |  | 4+1 = 5 |  |  |  |  |
| 02 |  | compteur<= 4 ? FAUX |  |  |  |  |  |  |  |  |  |
| 05 |  |  |  |  |  |  |  |  |  |  | Terminé ! |
| **FIN** |  |  |  |  |  |  |  |  |  |  |  |

# Exemple : Calculer la quantité totale d’un ensemble de produits

On souhaite avoir une procédure qui permette la saisie de quantité de plusieurs quantités d’un produit et d’en afficher la somme. On arrêtera la saisie quand la quantité saisie sera 0.

DECLARATIONS

VARIABLES

 qte : réel 🡨 0 // quantité, saisi

 total : réel 🡨 0 // somme, saisi

DEBUT (des opérations)

// demander la saisie d'une 1ere quantité

 AFFICHER « donnez une quantité (0 pour arrêter) : »

 SAISIR qte

// cumul des quantités

 **TANTQUE** (qte <> 0)

**REPETER**

 // ajouter la quantité saisie au total

 total 🡨 total + qte

 // saisir une autre quantité

 AFFICHER « donnez une autre quantité (0 pour arrêter) : »

 SAISIR qte

 **finTANTQUE**

// donner le résultat

 AFFICHER « le total des quantités est  », total

FIN (des opérations)

Effectuer une trace d'exécution pour les valeurs successives saisies :

* essai1 : 0
* essai2 : 1 4 0

Compléter la procédure afin de saisir aussi le prix unitaire, et afficher également le montant total.

Les clubs versent chaque année des cotisations (clubs et adhérents) auprès de la FFTDA. Les cotisations payées par les clubs et les adhérents permettent de couvrir une partie des frais de fonctionnement :

* des clubs (location et gestion des salles, organisation d’évènements tels que des compétitions locales) ;
* des ligues régionales et des comités départementaux (championnats régionaux, compétitions régionales, passages de grades…) ;
* de la FFTDA, qui perçoit à ce titre de 30 € ou 34 € par adhérent selon qu'il ait plus ou moins 18 ans, 180 € par club, 40 € par membre de chaque bureau des ligues régionales ainsi que 2 € par adhérent au titre de la prime annuelle d’assurance.

|  |  |
| --- | --- |
| Nombre de clubs | % de réduction sur la cotisation annuelle hors assurance |
| 1 à 15 clubs | 5 % |
| 16 à 19 clubs | 3 % |
| 20 clubs et plus | 1,5 % |