

Les structures de contrôle

Les structures de contrôle conditionnelles

L'exécution conditionnelle permet de n'exécuter une instruction que si une certaine condition est remplie

Syntaxe :

- ***Si*** Condition ***Alors***
Instruction;
FinSi
- ***Si*** Condition ***Alors***
Instruction1
Sinon
Instruction2;
FinSi
- ***Si*** Condition1 ***Alors***
Instruction1
Sinon Si condition2 ***Alors***
Instruction2
Sinon
Instruction 3;
FinSi

Les conditions

La condition est une expression dont le type est booléen :

- VRAI
- FAUX

La condition comprend deux forme :

- Une variable

Exemple : Si a Alors \Leftrightarrow Si a == VRAI Alors

- Comparaison

Exemple : Si a > b Alors

Les comparaisons simples

Une comparaison simple comprend trois éléments :

- Une valeur/variable
- Un opérateur de comparaison
- Une valeur/variable

Exemple :

- Si $a > 2$ Alors
- Si $a == b$ Alors

Les comparaison composées

Une comparaison composée comprend plusieurs conditions simple reliées par des opérateurs logiques

Une comparaison composée comprends :

- Une comparaison simple
- Un opérateur logique
- Une comparaison simple

Exemple :

- Si $a > 2$ ET $b < 3$ Alors
- Si $'t' == 'w'$ OU $'a' < 'b'$ Alors

Exercice 6

Ecrire un algorithme qui demande un nombre à l'utilisateur, et l'informe ensuite si ce nombre est positif ou négatif (on laisse de côté le cas où le nombre vaut zéro).

Exercice 6

Algorithme Positif_Negatif;

Variable n : Entier;

Début

 Écrire ("Veuillez saisir un nombre");

 Lire (n);

 Si $n \geq 0$ Alors

 Écrire ("Ce nombre est positif")

 Sinon

 Écrire ("Ce nombre est négatif");

 Finsi

Fin

Exercice 7

Ecrire un algorithme qui demande deux nombres à l'utilisateur et l'informe ensuite si leur produit est négatif ou positif (on laisse de côté le cas où le produit est nul). Attention toutefois : on ne doit pas calculer le produit des deux nombres.

Exercice 7

Algorithme Produit_Postif_Negatif

Variables m, n : Entier;

Début

 Ecrire ("Veuillez introduire deux nombres");

 Lire (m, n);

 Si $(m > 0 \text{ ET } n > 0) \text{ OU } (m < 0 \text{ ET } n < 0)$ Alors

 Écrire ("Leur
produit est positif ")

 Sinon

 Écrire ("Leur
produit est négatif ");

 Finsi

Fin

Les structures de contrôle répétitives

Une boucle est une instruction qui permet de répéter l'exécution d'une partie d'un programme

On distingue trois types de boucles :

- La boucle Pour
- La boucle TantQue
- La boucle Répéter

Syntaxe :

- **Pour** Indice = Val1 à Val2 **Faire** :
 Instructions;
 FinPour
- **TantQue Condition Faire**
 Instructions;
 FinTanQue
- **Répéter**
 Instructions;
 Jusqu'à Condition;

La boucle Pour

- Cette structure s'utilise si le nombre d'itération est connu
- Elle utilise une variable dite compteur défini par :
 - Sa valeur initiale
 - Sa valeur finale
 - Son pas de variation
- Lorsque le nombre d'itérations à réaliser est atteint, l'exécution des instructions de la boucle est interrompu

Exercice 8

Écrire un algorithme qui demande à un étudiant de saisir ses notes (5 modules) puis les afficher

Exercice 8

Correction : Solution proposé sans utilisation de boucles

Algorithme Note;

Variable N1,N2, N3, N4, N5 : Réel;

Début

Écrire ("Veuillez saisir votre note 1");

Lire (N1);

Écrire ("Veuillez saisir votre note 2");

Lire (N2);

Écrire ("Veuillez saisir votre note 3");

Lire (N3);

Écrire ("Veuillez saisir votre note 4");

Lire (N4);

Écrire ("Veuillez saisir votre note 5");

Lire (N5);

Écrire(" Votre note 1 =", N1);

Écrire(" Votre note 2 =", N2);

Écrire(" Votre note 3 =", N3);

Écrire(" Votre note 4 =", N4);

Écrire(" Votre note 5 =", N5);

Fin

Exercice 8

Correction : Solution proposé en utilisant la boucle Pour

Algorithme Note;

Variable N : Réel;

i : Entier;

Début

Pour i=1 à 5 Faire :

Écrire ("Veuillez saisir votre note ", i);

Lire (N);

Écrire(" Votre note ", i, " = ", N);

FinPour

Fin

Les structures de contrôle répétitives

Une boucle est une instruction qui permet de répéter l'exécution d'une partie d'un programme

On distingue trois types de boucles :

- La boucle Pour
- La boucle TantQue
- La boucle Répéter

Syntaxe :

- **Pour** Indice = Val1 à Val2 **Faire** :
 Instructions;
 FinPour
- **TantQue Condition Faire**
 Instructions;
 FinTanQue
- **Répéter**
 Instructions;
 Jusqu'à Condition;

La boucle TantQue

- La boucle TantQue permet de répéter l'exécution d'un bloc d'instructions tant que la condition est vrai
- La boucle TantQue permet de répéter l'exécution d'un bloc d'instruction jusqu'à ce que la condition soit faux

Exercice 9

Ecrire un algorithme qui demande à l'utilisateur un nombre positif jusqu'à ce que la réponse convienne.

Exercice 9

```
Algorithme Nbr_Positif;  
Variable N : Entier;  
Début  
    Écrire ("Nombre positif?")  
    Lire (N);  
    TantQue N ≤ 0 Faire  
        Écrire ("Nombre positif?");  
        Lire (N);  
    FinTantQue  
Fin
```

Les structures de contrôle répétitives

Une boucle est une instruction qui permet de répéter l'exécution d'une partie d'un programme

On distingue trois types de boucles :

- La boucle Pour
- La boucle TantQue
- La boucle Répéter

Syntaxe :

- **Pour** Indice = Val1 à Val2 **Faire** :
 Instructions;
 FinPour
- **TantQue Condition Faire**
 Instructions;
 FinTanQue
- **Répéter**
 Instructions;
 Jusqu'à Condition;

La boucle Répéter

- Cette boucle permet de répéter l'exécution d'un bloc d'instruction jusqu'à ce que la condition soit vraie

Exercice 10

Ecrire un algorithme qui demande à l'utilisateur un nombre positif jusqu'à ce que la réponse convienne.

Exercice 10

Algorithme Nbr_Positif;

Variable N : Réel;

Début

 Répéter

 Écrire ("Nombre positif?");

 Lire (N);

 Jusqu'à ($N > 0$);

Fin

TantQue Vs Répéter

les structures RÉPÉTER-JUSQU'À et TANTQUE diffèrent sur deux points :

- La structure TANTQUE exécute la séquence d'instructions tant et aussi longtemps que la condition est satisfaite
- La structure RÉPÉTER-JUSQU'À exécute la séquence d'instructions tant et aussi longtemps que la condition n'est pas satisfaite.
- La structure TANTQUE vérifie la condition avant chaque itération
- La structure RÉPÉTER-JUSQU'À vérifie la condition après chaque itération : la séquence d'instructions est exécutée au moins une fois, sans égard à la valeur de la condition

Exercice 11

Écrire un algorithme qui demande un nombre compris entre 10 et 20, jusqu'à ce que la réponse convienne. En cas de réponse supérieure à 20, on fera apparaître un message : « Plus petit ! », et inversement, « Plus grand ! » si le nombre est inférieur à 10.

Exercice 11

Algorithme Nombre_10_20;
Variable N : Entier;

Debut

 N=0;

 Écrire ("Entrez un nombre entre 10 et 20 »);

 TantQue (N < 10 ou N > 20) faire

 Lire (N);

 Si (N < 10) Alors

 Écrire ("Plus grand! »)

 Sinon Si (N > 20) Alors

 Écrire ("Plus petit! ");

 FinSi

 FinTantQue

Fin

Exercice 12

Écrire un algorithme qui calcule puis affiche la moyenne d'une liste de n valeurs fournies par l'utilisateur au clavier de la machine. L'utilisateur fournit la valeur de n , puis la liste des n valeurs à considérer.

Exercice 12

Algorithme Moyenne;

Variable nbr, N, S, i :Entier;

 M : Réel;

Début

 Écrire ("Veuillez saisir le nombre de valeur");

 Lire (nbr);

 i=0;

 S=0;

 Pour i=1 à nbr faire :

 Écrire("Veuillez saisir un nombre") ;

 Lire(N) ;

 S=S+N ;

 FinPour

 M=S/nbr ;

 Écrire ("La moyenne = ", S) ;

Fin

Exercice 13

Écrire un algorithme qui calcule, puis affiche la moyenne d'une liste de valeurs fournies par l'utilisateur au clavier de la machine. L'utilisateur fournit les valeurs à traiter et indique la fin de la liste par l'entrée de la valeur 0. Évidemment cette valeur 0 ne fait pas partie de la liste des valeurs à considérer.

Exercice 13

Algorithme Moyenne;
Variable N, S , M : Réel;
i : Entier ;

Début

S=0;

i=0;

Répéter

Écrire ("veuillez saisir un nombre");

Lire (N);

S = S+N;

i=i+1;

Jusqu'à N==0;

M= S/(i-1);

Écrire("la moyenne =", M);

Fin

Exercice 14

Ecrire un algorithme qui demande à l'utilisateur un nombre compris entre 1 et 3 jusqu'à ce que la réponse convienne.

Exercice 14

Algorithme Nombre_1_3;

Variable N : Entier;

Début

Écrire ("Entrez un nombre entre 1 et 3 ");

Lire (N);

TantQue ($N \leq 1$ ou $N \geq 3$) faire

Écrire ("Saisie erronée, veuillez entrez un nombre entre 1 et 3 ");

Lire (N);

FinTantQue

Fin

Exercice 14

Algorithme Nombre_1_3;

Variable N : Réel;

Début

 Répéter

 Écrire (" Veuillez entrez un nombre entre 1
 et 3 ");

 Lire (N);

 Jusqu'à (N > 1 ET N < 3);

Fin