

Cours: Introduction à l'algorithmique



Pr. Hajar IGUER,
hajar.iguer@uic.ac.ma



LES TESTS: STRUCTURES CONDITIONNELLES



Sélective Selon

- L'instruction conditionnelle à choix multiple **Selon** permet de choisir d'exécuter une action en fonction de la valeur d'un sélecteur.
- Elle évalue l'expression et n'exécute que les instructions qui correspondent à la valeur du sélecteur.



Sélective Selon

- Le **sélecteur** peut être une variable ou une expression arithmétique ou logique..
- Le **sélecteur** doit avoir le même type que les valeurs devant les cas.
- L'instruction **Selon** permet une facilité d'écriture.
- La clause Cas Autre est facultative et permet de traiter tous les cas non traités précédemment.



Sélective Selon: Syntaxe

Selon sélecteur

Cas valeur1 :

instructions1;

Cas valeur2 :

instructions2;

...

Cas valeurN :

instructionsN;

Cas Autre

instructions.

FinSelon



Selon..... Fin Selon: Exemples

- Cas 1, 2 veut dire valeur de l'expression égale à 1 ou 2
- Cas > 3 veut dire valeur de l'expression supérieure à 3
- Cas 11 à 20 veut dire valeur de l'expression entre 11 et 20 (inclus)
- Reproduire les tests imbriqués dans la syntaxe SELON ...FINSELON



Exercice: Selon..... Fin Selon

- Ecrire l'algorithme qui permet de saisir un numéro de couleur de l'arc-en-ciel et d'afficher la couleur correspondante :
1 : rouge, 2 : orangé, 3 : jaune,
4 : vert, 5 : bleu, 6 : indigo et 7 : violet.



Exercice: Structures conditionnelles

- Ecrivez un algorithme qui affiche selon la quantité achetée d'un produit X, le prix au Kg de ce produit.

quantité	prix au kg
entre 1 et 5 kg (exclus)	100
entre 5 et 10 kg (exclus)	90
entre 10 et 30 kg (exclus)	80
entre 30 et 60 kg (exclus)	70
à partir de 60 kg	55



LES STRUCTURES RÉPÉTITIVES OU ITERATIVES



Structure répétitive

- Une **structure répétitive ou itérative** reprend l'exécution d'un traitement donné.
- Il existe plusieurs formes d'utilisations d'une structure itérative
 - Lorsqu'on connaît à l'avance le nombre d'itérations
 - Lorsqu'on exécute un traitement si une condition a été vérifiée
 - Lorsqu'on exécute un traitement une première fois avant de vérifier une condition



BOUCLES

- Lorsqu'on a plusieurs actions qui se répètent, on écrit celles-ci dans une même action composée et l'on réitère plusieurs fois son exécution.
- Le nombre d'itérations peut être connu a priori ou pas.
- Dans ce dernier cas c'est l'exécution de l'action itérative qui déterminera son arrêt.
- Il existe plusieurs façons d'exprimer une action itérative.



Tant que: Syntaxe

***Tant que Condition
faire***

...

ACTION

...

Fin Tant que

- Tant que la condition ou booléen est vérifiée ou VRAI on exécutera le corps de la boucle,
- On s'arrêtera dès que la condition n'est plus vérifiée ou la condition est de valeur FAUSSE.



Exemple 1

Algorithme A1

Variable A : Entier;

$A \leftarrow 0$;

TantQue A < 15

$A \leftarrow A + 1$;

Ecrire "Passage

numéro : ", A);

FinTantQue

- Attention de mettre dans la boucle Tant que une condition toujours vérifiée.
- L'ordinateur ne sort jamais de la boucle infinie.



Exemple 2

Algorithme Moyenne ;

Variable Note, Moyenne, Som : réel ;

Variable nb : entier ;

Début

Som \leftarrow 0; nb \leftarrow 0 ;

Lire(' Entrer une note : ', Note) ;

Tant que Note \geq 0 faire

nb \leftarrow nb + 1 ;

Som \leftarrow Som + Note ;

Lire(' Entrer une note : ', Note) ;

Fin Tant que

Moyenne \leftarrow Som / nb;

Ecrire(' La moyenne des ', nb ' notes est de : ', Moyenne);

Fin



Exercices

- Réécrivez l'algorithme précédent (arc-en-ciel) en utilisant la structure itérative tant que
- Ecrire un algorithme qui affiche les dix nombres suivants d'une valeur numérique saisie au clavier.
- Ecrire un algorithme qui affiche le résultat de la suite arithmétique suivante:
- $un = \sum_{i=0}^n 2^i$



La boucle Répéter

- La boucle Répéter permet de rentrer dans la boucle quelque soit la condition et réitère l'exécution jusqu'à ce que la condition soit vérifiée.
- Dans cette boucle, le traitement est exécuté au moins une fois avant l'évaluation de la condition d'arrêt.
- Il doit y avoir une action dans le traitement qui modifie la valeur de la condition.



La Boucle Répéter: Syntaxe

Répéter

...

ACTION

...

Jusqu'à Condition

- Exécuter les instructions jusqu'à ce que la condition soit vraie.
- Lorsque la condition devient vraie, l'exécution continue après l'instruction « jusqu'à condition ».



Exemple

Algorithme Moyenne

Variable Note, Moyenne, Som : réel ;

Variable nb : entier ;

Début

Som \leftarrow 0; nb \leftarrow 0 ;

Répéter

Ecrire(' Entrer une note : ')

Lire(Note) ;

nb \leftarrow nb +1 ;

Som \leftarrow Som + Note ;

Jusqu'à Note \leftarrow 0

Moyenne \leftarrow Som / nb;

Ecrire(' La moyenne des ', nb ' notes est de : ', Moyenne);

Fin



La Boucle POUR

- La boucle **POUR** est utilisée lorsqu'on connaît le nombre de répétition du traitement d'avance.
- L'usage principal de la boucle **POUR** est de faire la gestion d'un compteur (de type entier) qui évolue d'une valeur à une autre.
- La valeur du pas est égale à 1 par défaut et peut être positif ou négatif.



La Boucle POUR: Syntaxe

POUR variable **de** valeur1 **À** valeur2

Pas ValeurDuPas

Instructions;

Fin Pour



La Boucle POUR

- La variable, jouant le rôle d'un compteur, est initialisée à la **valeur1**.
- L'ordinateur teste si la variable est inférieure ou égale à la **valeur2**.
- si c'est le cas, l'instruction ou le bloc d'instruction est effectué, la variable jouant le rôle de compteur est augmentée de 1, et retour à l'étape de l'exécution des instructions.



La Boucle POUR

- Si ce n'est pas le cas, l'instruction ou le bloc d'instruction n'est pas effectuée, et l'ordinateur passe aux instructions après la fin de la boucle **POUR**.
- Le seul intérêt du « **Pour** » est d'épargner un peu de fatigue au programmeur, en lui évitant de gérer lui-même la progression de la variable qui lui sert de compteur.



La Boucle POUR

- La structure « Pour ... Fin Pour » est un cas particulier de TantQue : celui où le programmeur peut dénombrer à l'avance le nombre de tours de boucles nécessaires.
- Lorsque la valeur du pas est 1 on évite de le mentionner dans la boucle Pour.



La Boucle POUR: Exercices

- Réécrivez les deux derniers exercices (les dix nombres suivants ,suite arithmétique) de tant que avec la boucle Pour.....Fin Pour.
- Ecrire l'algorithme qui permet de saisir les moyennes des N étudiants de la classe Informatique et de calculer la moyenne générale de la classe.



La Boucle POUR: Exercices

- Ecrire l'algorithme qui permet d'afficher tous les nombres impairs entre 50 et 100 dans l'ordre décroissant.
- Ecrire un algorithme qui lit les caractères saisis par l'utilisateur. A la fin ce programme nous affichera la phrase saisie. La saisie des caractères s'arrête lorsqu'on tape point «. ». Pour l'utilisateur veut insérer un espace il lui suffit de taper sur 0. Par exemple si l'utilisateur tape successivement les caractères « b », « o », « n », « j », « o », « u », « r », « t », « o », « u », « s », «.», il nous affichera la chaîne « bonjour tous ».
- Mais si il tape « b », « o », « n », « j », « o », « u », « r », « 0 », « t », « o », « u », « s », « . » , le programme affichera « bonjour tous ».



Remarque

- Si le nombre d'itérations est connu à l'avance,
 - Choisir la boucle **Pour....fin pour**
- Si la boucle doit s'arrêter quand survient un évènement ,
 - Choisir la boucle **Tant que.....fin tant que**
- Si la boucle doit s'exécuter une fois avant subir la condition d'un événement
 - Choisir la boucle **Répéter....jusqu'à**

