

# INSTITUT SUPÉRIEUR D'ENSEIGNEMENT PROFESSIONNEL DE DIAMNIADIO FILIÈRE TIC

## ALGORITHME INTRODUCTION

2021-2022

Présenté par: Abdoulaye MBAYE

+221772389823

[ambaye@isepdiamniadio.edu.sn](mailto:ambaye@isepdiamniadio.edu.sn)

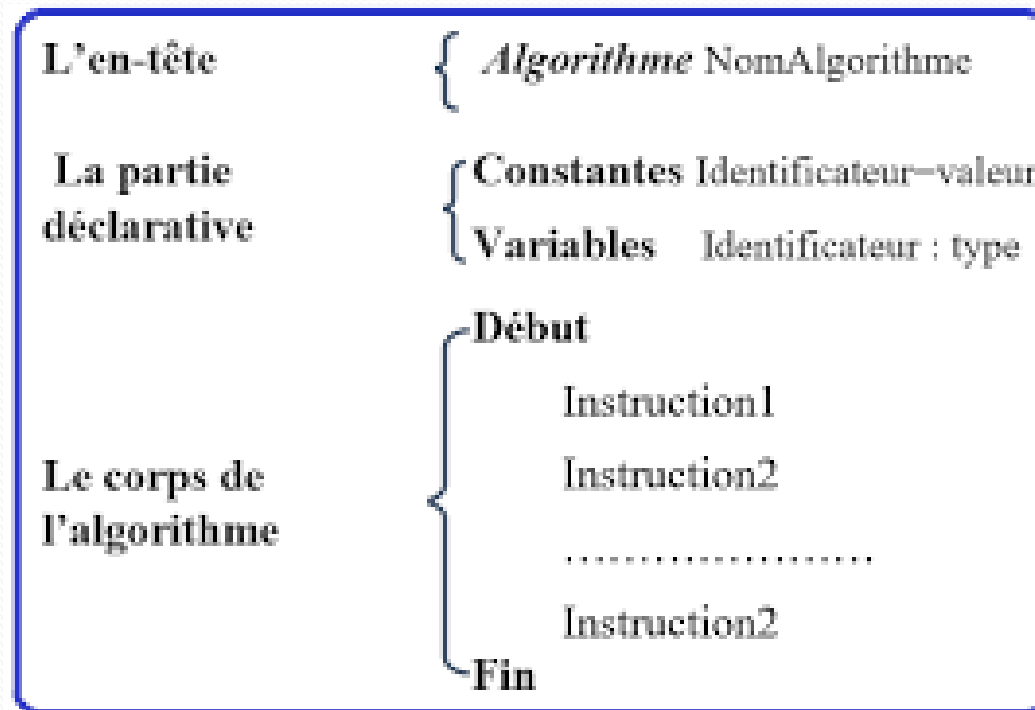
# PLAN

- **STRUCTURE D'UN ALGORITHME**
- **LES STRUCTURES DE CONTRÔLE**
  - **STRUCTURES ALTERNATIVE**

# Structure d'un algorithme

- Un algorithme (comme un programme) est composé de trois parties principales :
  1. La partie **définitions** permet de définir les «objets» qui pourront être manipulées dans l'algorithme. En particulier, on définit des constantes, des types et des sous-programmes.
  2. La partie **déclarations** permet de déclarer les données qui sont utilisées par le programme. Les données sont représentées par des variables.
  3. La partie **instructions** constitue le programme principal. Les instructions sont exécutées par l'ordinateur pour transformer les données.

# Structure d'un algorithme



# Structure d'un algorithme



# Structure d'un algorithme

- **Exemple:**

**Algorithme:** calcul

**Variable**

A: Entier  
C, B : Réel  
D : Chaîne  
E : Booléen

**Début**

A  $\leftarrow$  30  
B  $\leftarrow$  A \* 2  
Ecrire ("B = ", B)  
C  $\leftarrow$  (B+A)/4  
B  $\leftarrow$  C / 5  
D  $\leftarrow$  "Amine"  
E  $\leftarrow$  (A > 40) OU (C < B)  
Ecrire("Les valeurs obtenues sont : A = ", " B = ", B, " C = ", C, "D = ", D, "E = ", E)

**FIN**

# LES STRUCTURES DE CONTRÔLE

- En programmation impérative, une **structure de contrôle** est une commande qui contrôle l'ordre dans lequel les différentes instructions d'un algorithme ou d'un programme informatique sont exécutées.
- On appelle aussi cet enchaînement d'instructions le flot d'exécution d'un programme.
- Un programme s'arrête généralement après l'exécution de la dernière instruction.

# STRUCTURES ALTERNATIVE CONDITIONNELLE

- La conditionnelle permet d'exécuter une séquence d'instruction, seulement si une condition est vraie.
- Elle permet aussi à un programme de modifier son traitement en fonction d'une condition.
- Il existe trois forme d'instruction conditionnelles:
  - Forme simple
  - Forme généralisée
  - Forme à choix



# CONTRÔLE

## CONDITIONNELLE SIMPLE

- Une structure de contrôle conditionnelle est dit forme simple réduite lorsque le traitement dépend d'une condition. Si la condition est évaluée à « vraie », le traitement est exécuté.

- Syntaxe

- **Si**(expression logique(vaire)) **alors**

Instruction 1

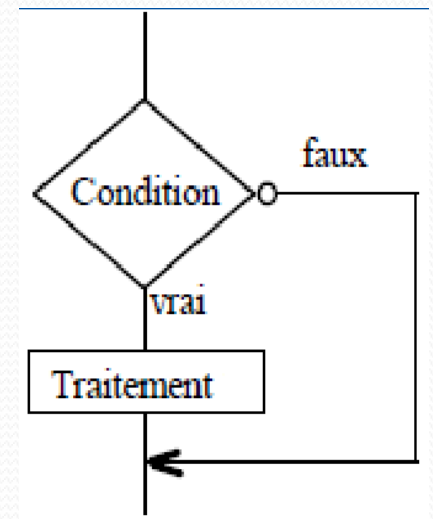
Instruction 2

....

Instruction N

**FinSi**

La condition (expression logique) est de type booléen



# CONTRÔLE

## CONDITIONNELLE SIMPLE

- **Exemple:** Ecrire un algorithme qui permet de faire la racine carrée d'un réel  $x$ .
  - **Algorithme:** racineCarrée  
Variable  $x$ : réel  
Début  
    Ecrire("Donner un réel")  
    Lire( $x$ )  
    **Si** ( $x \geq 0$ ) **alors**  
        Ecrire ("La racine carrée est = ",  $\text{sqrt}(x)$ )  
    **FinSi**  
Fin

# CONTRÔLE

## CONDITIONNELLE SIMPLE

- **Exemple:** Ecrire un algorithme qui permet de vérifier si un entier donné est paire (en utilisant la forme simple).

- **Algorithme:** parité

Variable n: Entier

message : Chaine

Début

Ecrire("Donner un entier")

Lire(n)

message ← "Pair"

**Si** ( $n \bmod 2 \neq 0$ ) **alors**

message ← "impair"

**FinSi**

Ecrire(n, " est nombre ", message)

Fin

# CONTRÔLE

## CONDITIONNELLE SIMPLE

- Une structure de contrôle conditionnelle est dite à forme alternative lorsque le traitement dépend d'une condition à deux état: Si la condition est évaluée à « vraie », le premier traitement est exécuter, si la condition est évaluée à « faux », le second traitement est exécuté.

- Syntaxe:

**Si** (condition « vraie ») **Alors**

Instruction 1 de TR<sub>1</sub>

Instruction 2 de TR<sub>1</sub>

.....

Instruction m de TR<sub>1</sub>

**Sinon**

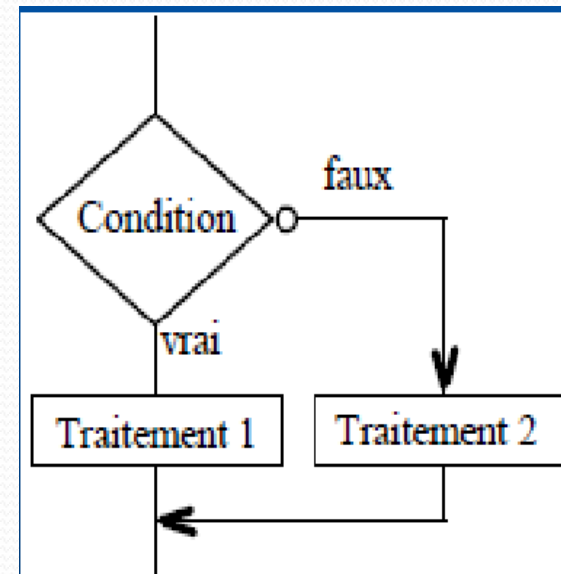
Instruction 1 de TR<sub>2</sub>

Instruction 2 de TR<sub>2</sub>

.....

Instruction n de TR<sub>2</sub>

**FinSi**



# CONTRÔLE

## CONDITIONNELLE SIMPLE

- Exemple 2: nombre paire
- Algorithme : Parité2

Variable **n**: entier

Début

**écrire**("Donner un entier ")

**lire** (n)

**si**  $n \bmod 2 = 0$  **alors**

**écrire**(n, " est pair")

**sinon**

**écrire**(n, " est impair")

**FinSi**

**Fin**

# CONTRÔLE CONDITIONNELLE GENERALISEE

- Une structure de contrôle conditionnelle est dite généralisée lorsqu'elle permet de résoudre des problèmes comportant plus de deux traitements en fonction des conditions. L'exécution d'un traitement entraîne automatiquement la non-exécution des autres traitements.
- Syntaxe:  
    **Si** condition 1 **Alors**  
        Traitement 1  
    **Sinon Si** condition 2 **Alors**  
        traitement 2  
    **Sinon Si** condition 3 **Alors**  
        traitement 3  
    .....  
    **Sinon Si** condition N-1 **Alors**  
        traitement N-1  
    **Sinon**  
        traitement N  
    **FinSi**

# CONTRÔLE CONDITIONNELLE GENERALISEE

- **Remarque :**
  - Il est préférable de mettre les événements les plus probables en premier lieu.
  - Chaque traitement peut comporter une ou plusieurs instructions.
- **Exemple:** Écrire un programme qui permet de faire la résolution de l'équation du 1er degré :  $ax+b = 0$

# CONTRÔLE CONDITIONNELLE GENERALISEE

- **Algorithme:** Equation

Variable a, b: réel

Début

Ecrire (" Donner a : ")

lire(a)

Ecrire ("Donner b : ")

lire(b)

Si (a ≠ 0) alors

$S \leftarrow -b/a$

Ecrire ("La solution est ", S)

sinon

si (b=0) alors

Ecrire("La solution est IR ")

Finsi

Sinon

Ecrire("La solution est vide ")

FinSi

Fin



# CONTRÔLE CONDITIONNELLE GENERALISEE

- **Exercice 1:**

Écrire un programme qui permet de saisir une moyenne (moy) puis affiche la décision correspondante :

ADMIS Si  $\text{moy} \geq 10$

CONTROLE Si  $9 \leq \text{moy} < 10$

REDOUBLE Si  $\text{moy} < 9$

- **Exercice 2:**

Écrire un programme qui permet de saisir un temps (heure et minute) lui ajoute 5 minutes puis l'affiche.

Exemples :

heure : 10

minute : 20

après 5 minutes : **10:25**

heure : 10

minute:57

après 5 minutes : **11: 02**

heure : 23

minute:55

après 5 minutes : **00:00**

## Remarque :

Pour afficher l'heure et minute correctement ( ajouter un zéro à gauche des chiffres  $< 10$  , exemple 05:08 )

# CONTRÔLE CONDITIONNELLE À CHOIX

- Une structure de contrôle conditionnelle est dite à choix lorsque le traitement dépend de la valeur que prendra le sélecteur, ce sélecteur doit être de type scalaire (entier ou caractère).

**Syntaxe:**

**selon sélecteur Faire**

**Valeur 1 : Action 1**

**Valeur 2 : Action 2-1**

**Action 2-2**

**Action 2-n**

**Valeur 3 : Action 3**

**Valeur 4, valeur 6, valeur 8: Action 4**

**Valeur 5, valeur 7, valeur 9 : Action 5**

**Valeur 10 .. Valeur 19 : Action 6**

**.....**

**Valeur N : Action N**

**Sinon**

**Action R**

**FinSelon**

# CONTRÔLE CONDITIONNELLE À CHOIX

- **Remarques** :pour le choix
  - Sinon est facultative
  - Un traitement qui comporte plusieurs instructions doit être délimité par début et Fin
  - Les valeurs du sélecteur sont de type scalaire(entier , booléen, caractère) ou type intervalle(entier, caractère)
- **Exercice 1:**

Écrire un programme qui permet de saisir le n° de mois puis affiche la saison correspondante.

Exemple : n°mois=7 affiche été

12,1,2	saison hiver
3,4,5	saison printemps
6,7,8	saison été
9,10,11	saison automne

# CONTRÔLE CONDITIONNELLE À CHOIX

- **Exercice 2:**

Écrire un programme intitulé TOUCHE, qui affiche selon le cas, la nature du caractère (consonne, voyelle, chiffre ou symbole) correspondant à une touche saisie. On considère que le clavier est verrouillé en minuscule.

- **Exercice 3**

Écrire un programme qui permet d'afficher le nombre de jour d'un mois donné.

**NB:** Une année bissextile est une année comptant 366 jours au lieu de 365, c'est-à-dire une année comprenant un 29 février (exemple 2012) sont bissextiles les années:

divisibles par 4 mais non divisibles par 100  
divisibles par 400.