



Département: Mathématiques & Informatique

Séance 3: Les structures conditionnelles

**Licence : Physique Chimie
Filières : Physique**

Pr: Youssef Ouassit

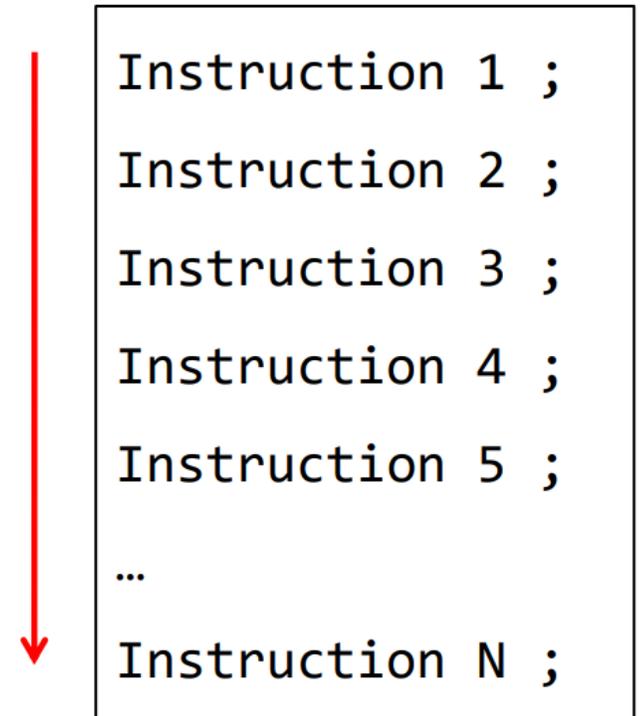
Les structures de contrôles :

- 1. Les structures conditionnelles (de choix)**
 - a. L'instruction conditionnelle simple
 - b. L'instruction conditionnelle alternative
 - c. Structure conditionnelle multiple

 - 2. Les instructions d'itération ou de répétition (les boucles)**
 - a) La boucle Pour ... Faire
 - b) La boucle Tant Que
- 

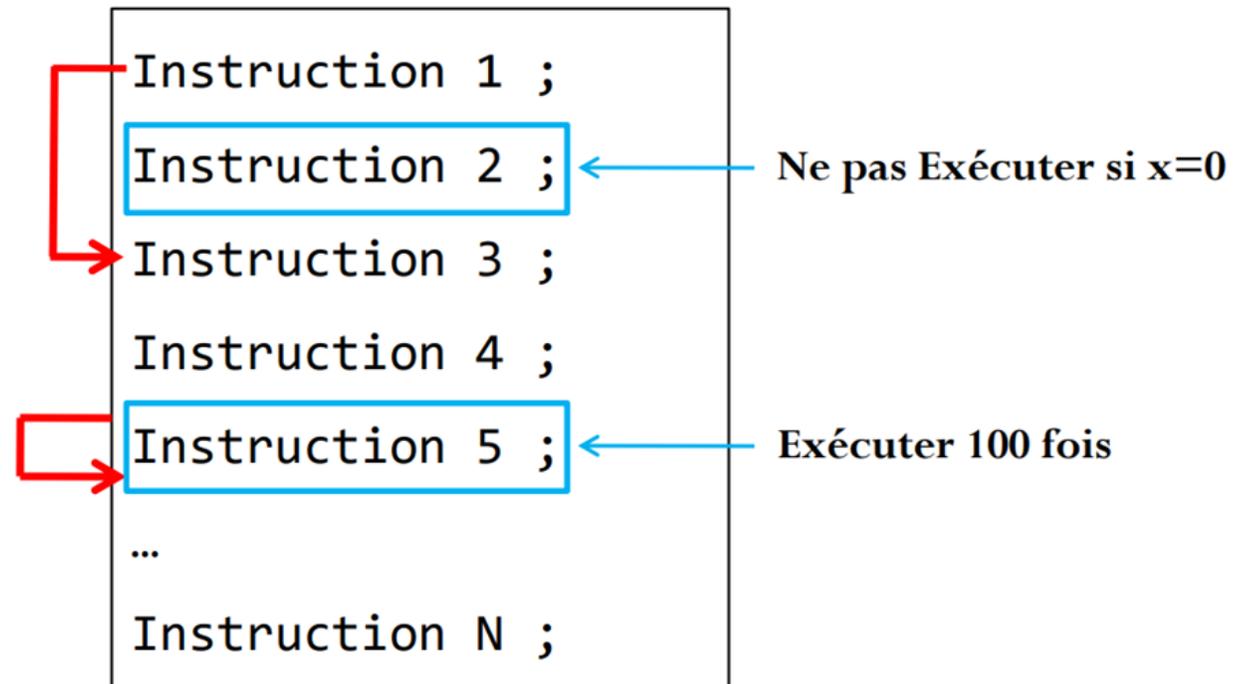
Structures séquentielles :

Une **structure algorithmique séquentielle** est la forme la plus simple d'organisation d'un programme où les instructions sont exécutées l'une après l'autre, dans l'ordre dans lequel elles sont écrites. Cette structure ne contient ni branches ni boucles; chaque étape est suivie par la suivante sans saut ni répétition.



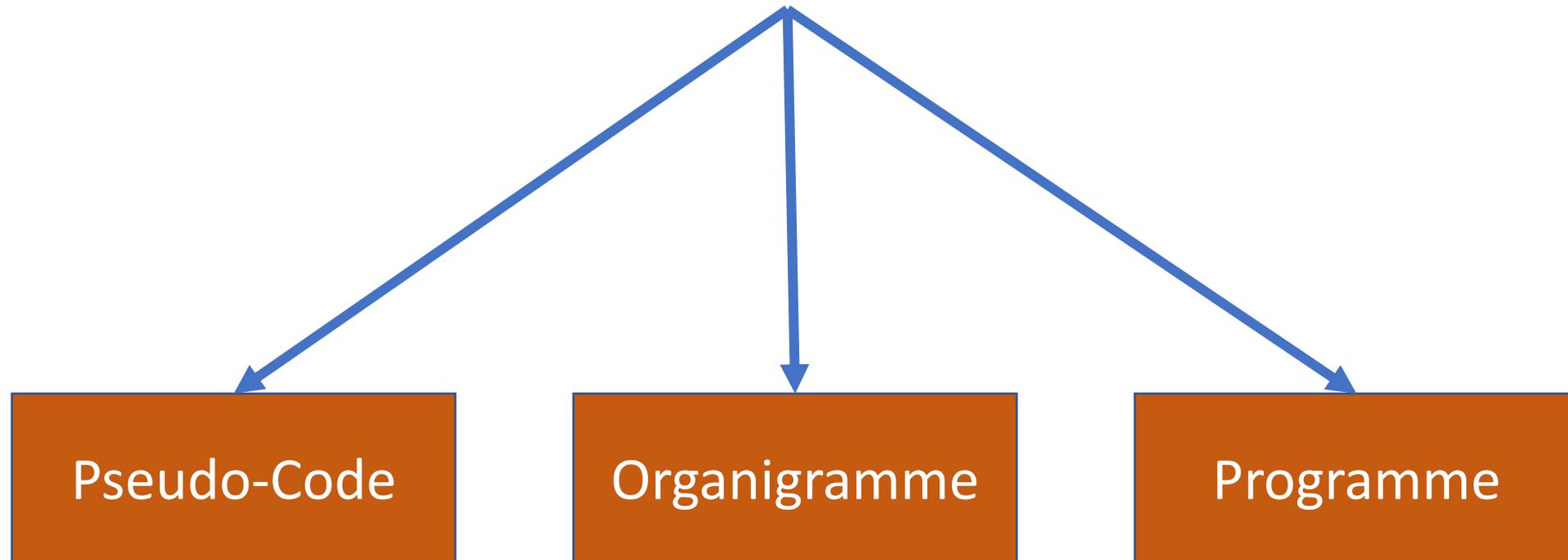
Dans Certains Cas, Nous avons besoin de modifier l'ordre d'exécution des instructions :

- Par exemple :
- Ne Pas Exécuter l' Instruction 2 si la variable x vaut 0.
 - Exécuter L' Instruction 5 100 fois



Structure Conditionnelle Alternative

Trois façons pour représenter un algorithme:



Un **organigramme d'algorithme** (ou diagramme de flux) est une représentation graphique des étapes d'un algorithme. Il utilise des symboles pour décrire les actions et les décisions prises au fil du déroulement de l'algorithme. Voici les principaux symboles utilisés dans un organigramme :

- 1) **Ovale** : Représente le début ou la fin de l'algorithme.
- 2) **Rectangle** : Représente une action ou un processus, comme une opération de calcul.
- 3) **Parallélogramme** : Représente une entrée ou une sortie (lecture ou écriture de données).
- 4) **Losange** : Représente une décision (ou un test conditionnel), avec deux flèches sortantes : une pour "vrai" et une pour "faux".

Organigramme

Algorithme Volume

Variables : R, H, V, SB : Réel

Constantes : Pi = 3.14

Début

Ecrire("Donner le rayon: ")

Lire(R)

Ecrire("Donner la hauteur: ")

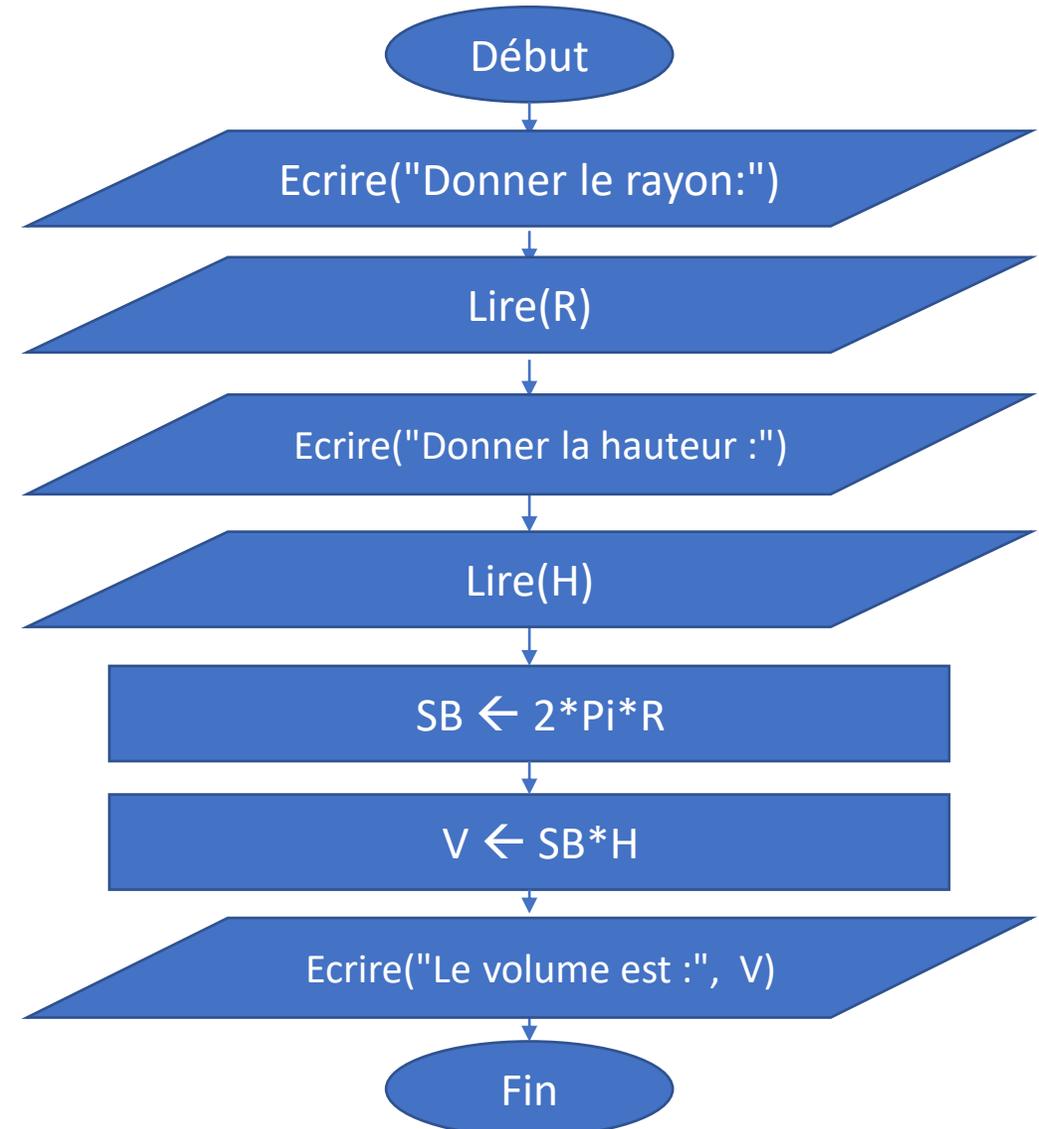
Lire(H)

$SB \leftarrow 2 * \text{Pi} * R$

$V \leftarrow SB * H$

Ecrire("Le volume est " , V)

Fin



Exemple 1:

Un algorithme qui calcul la division de deux nombres.

Algorithme Division

Variables : A, B : Réel

Début

Ecrire("Donner A: ")

Lire(A)

Ecrire("Donner B:")

Lire(B)

→ Ecrire("Le résultat : ", A/B)

→ Ecrire("Division par zéro")

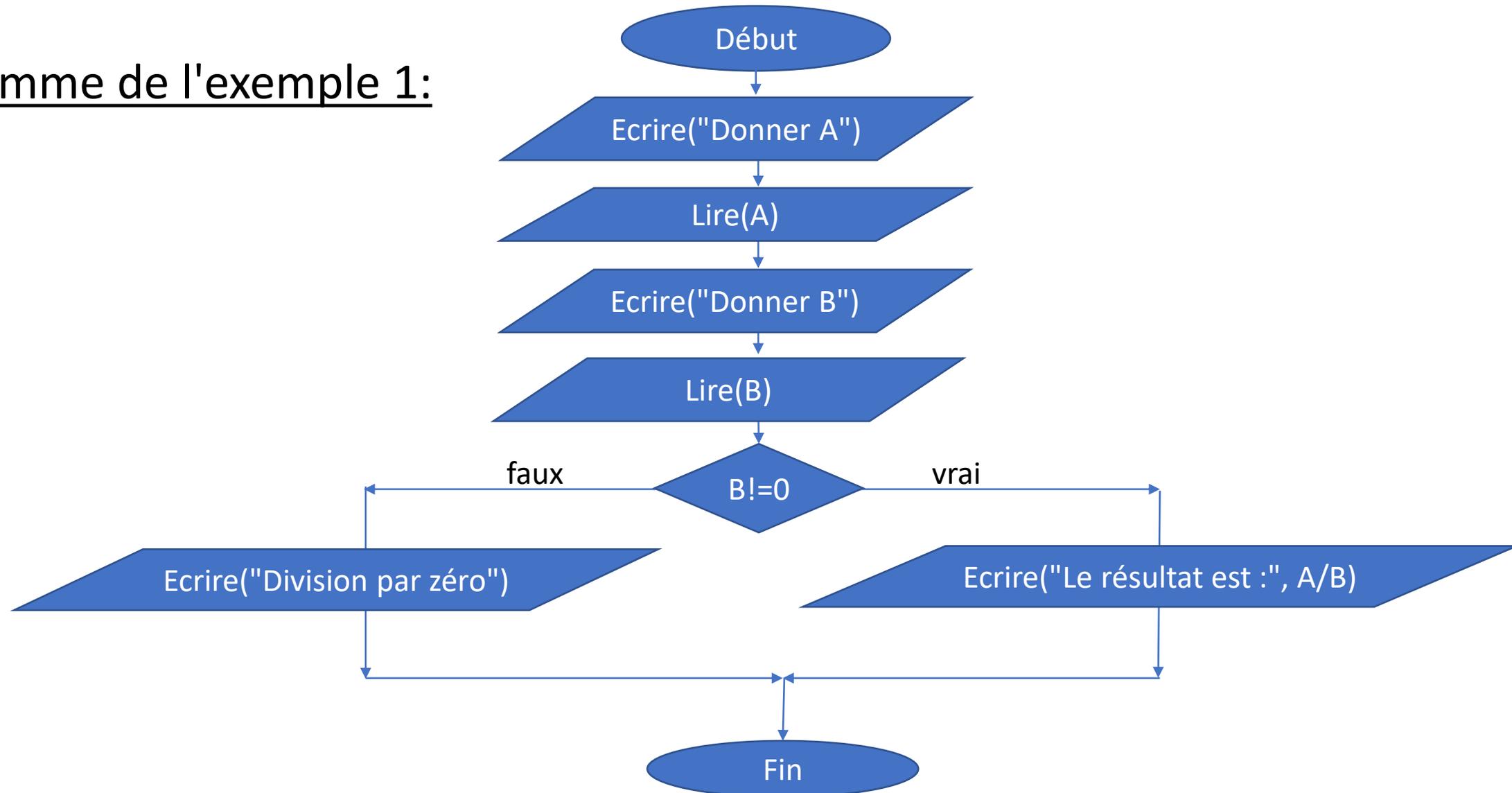
Fin



Structures conditionnelles

Structures conditionnelles alternatives

Organigramme de l'exemple 1:



Structures conditionnelles

Structures conditionnelles alternatives

Syntaxe :

Si condition Alors

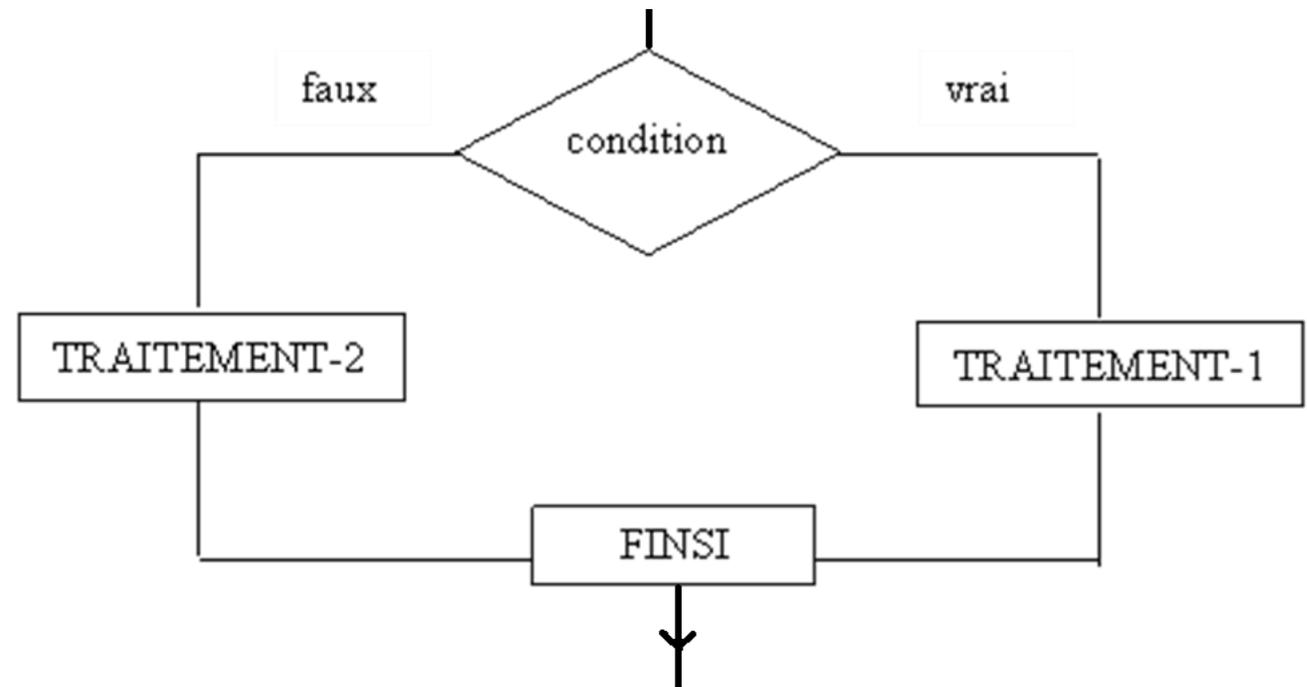
TRAITEMENT 1

SiNon

TRAITEMENT 2

FinSi

Organigramme:



Syntaxe :

Si $B \neq 0$ Alors

 Ecrire("Le résultat:", A/B)

SiNon

 Ecrire("Division par zéro")

FinSi

Algorithme de l'exemple 1:

Algorithme Division

Variables : A, B : Réel

Début

 Ecrire("Donner A: ")

 Lire(A)

 Ecrire("Donner B:")

 Lire(B)

 Si $B \neq 0$ Alors

 Ecrire("Le résultat:", A/B)

 SiNon

 Ecrire("Division par zéro")

 FinSi

Fin

Qu'est ce qu'une condition ?

Une condition est une expression booléenne, elle est soit vraie soit fausse.
La condition est soit simple, soit composée.

1- Condition simple : La condition simple est composée de deux opérandes et un opérateur de comparaison.

Les opérateurs de comparaison : `,` `<=`, `>=`, `=`, `<>`.

Exemples:

$$X = 2 * Y$$

$$X \leq 4$$

$$X \neq 0$$

$$A \bmod 2 = 0$$

2- Condition composée : La condition composée comporte plus qu'une condition simple qui sont reliées par des opérateurs logiques : ET, OU, NON.

Exemples :

- $A \in [0,20]$ est exprimé par : $(A \geq 0) \text{ ET } (A \leq 20)$
- $A \notin [0,20]$ est exprimé par : $(A < 0) \text{ OU } (A > 20)$
- $A \notin [0,20]$ est exprimé par : $\text{Non } ((A \geq 0) \text{ ET } (A \leq 20))$
- X est un multiple de 5 ou 7 : $(X \bmod 5 = 0) \text{ OU } (X \bmod 7 = 0)$

Exercice:

Soit x , y et z des entiers, Donner les expressions booléennes correspondant aux situations suivantes :

- x , y et z sont identiques.
 - $x=y$ et $y=z$
- x est pair
 - $x \bmod 2 = 0$
- x est impair.
 - $x \bmod 2 = 1$
- $x \in [y, z]$
 - $(A \geq 0)$ ET $(A \leq 20)$
- x est un multiple de y
 - $x \bmod y = 0$
- x est un nombre divisible par 5 mais pas par 10
 - $(x \bmod 5 = 0)$ ET $(x \bmod 10 \neq 0)$

Structure Conditionnelle Simple

Exemple 2: Un étudiant est accepté

Un Algorithme qui lit la note d'un étudiant et détermine si il est admis ou non.

Et si on ne s'intéresse qu'au cas des admis ?

Algorithme Admis

Variables : M : Réel

Début

Ecrire("Votre moyenne : ")

Lire(M)

Si $M \geq 10$ Alors

Ecrire("Vous êtes admis")

~~SiNon~~

~~Ecrire("Vous n'êtes pas admis")~~

FinSi

Fin

Structures conditionnelles

Structure conditionnelle simple

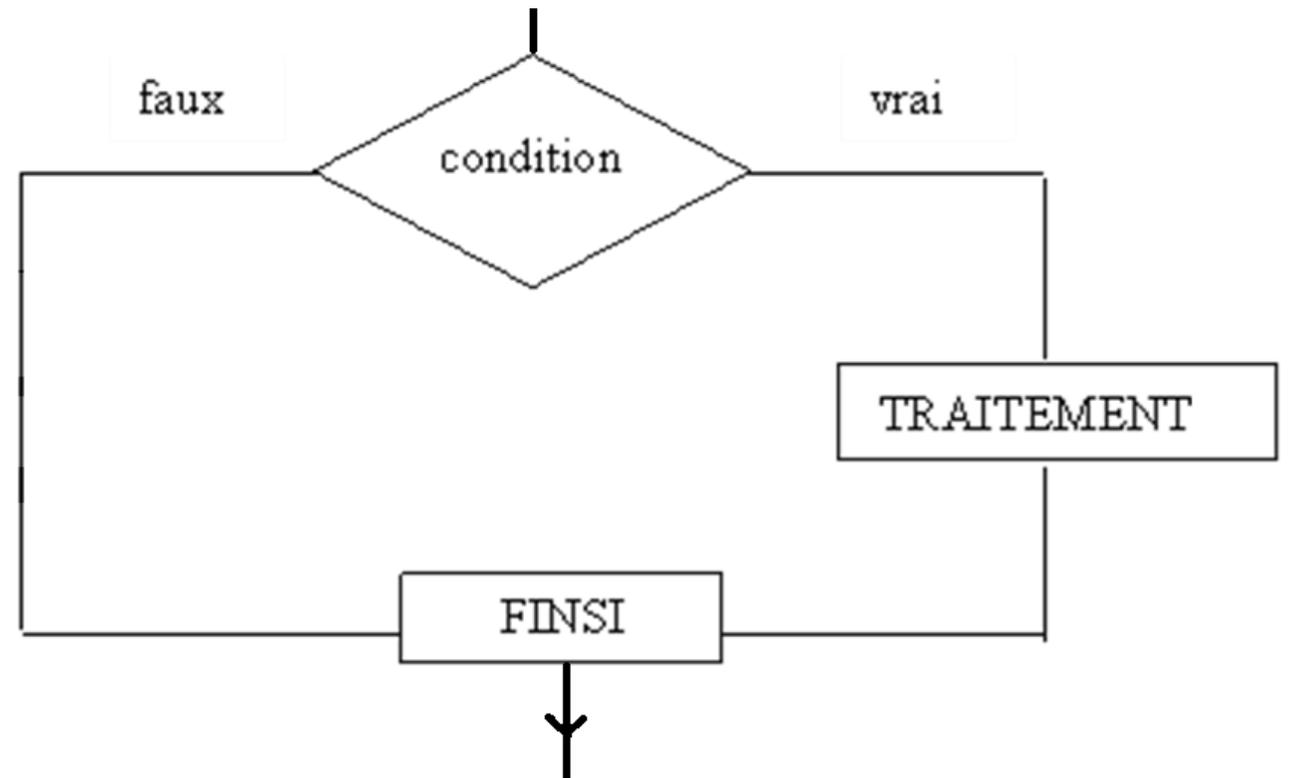
Syntaxe :

Si condition Alors

TRAITEMENT

FinSi

Organigramme:



Structures conditionnelles

Structure conditionnelle simple

Exemple 2 :

Algorithme Moyenne

Variables : M : Réel

Début

Ecrire("Votre moyenne : ")

Lire(M)

Si M \geq 10 Alors

Ecrire("Vous êtes admis")

SiNon

Ecrire("Vous n'êtes pas admis")

FinSi

Fin

Algorithme Moyenne

Variables : M : Réel

Début

Ecrire("Votre moyenne : ")

Lire(M)

Si M \geq 10 Alors

Ecrire("Vous êtes admis")

FinSi

Fin

Equivalence entre les deux syntaxes:

Alternative:

Si condition Alors

Traitement 1

SiNon

Traitement 2

FinSi



Simple:

Si condition Alors

Traitement 1

FinSi

Si Non(condition) Alors

Traitement 2

FinSi

Structures conditionnelles

Structure conditionnelle simple

Exemple 2: Avec des structures conditionnelles simple

Algorithme Moyenne

Variables : M : Réel

Début

Ecrire("Votre moyenne : ")

Lire(M)

Si $M \geq 10$ Alors

Ecrire("Vous êtes admis")

SiNon

Ecrire("Vous n'êtes pas admis")

FinSi

Fin

Algorithme Moyenne

Variables : M : Réel

Début

Ecrire("Votre moyenne : ")

Lire(M)

Si $M \geq 10$ Alors

Ecrire("Vous êtes admis")

FinSi

Si $M < 10$ Alors

Ecrire("Vous n'êtes pas admis")

FinSi

Fin

Exemple 3 :

Ecrire un algorithme qui affiche la valeur absolue d'un nombre, proposer deux solutions, une avec un SI alternative et une solution avec un SI simple.

Données d'entrés:

N : Réel

Données de sortie :

V : Réel

Traitement :

$V = N$ si $N \geq 0$

$V = -N$ si $N < 0$

Exemple 3 :

Avec SI alternative :

Algorithme Valeur Absolue

Variables : N, V : Réel

Début

Ecrire("Une nombre : ")

Lire(N)

Si N \geq 0 Alors

V \leftarrow N

SiNon

V \leftarrow - N

FinSi

Ecrire("La valeur absolue est : ", V)

Fin

Avec deux SI simples :

Algorithme Valeur Absolue

Variables : N, V : Réel

Début

Ecrire("Une nombre : ")

Lire(N)

Si N \geq 0 Alors

V \leftarrow N

FinSi

Si N < 0 Alors

V \leftarrow -N

FinSi

Ecrire("La valeur absolue est : ", V)

Fin

Exemple 3 :

Algorithme Valeur Absolue

Variables : N, V : Réel

Début

Ecrire("Une nombre : ")

Lire(N)

Si $N \geq 0$ Alors

$V \leftarrow N$

SiNon

$V \leftarrow -N$

FinSi

Ecrire("La valeur absolue est : ", V)

Fin

Avec une SI simple :

Algorithme Valeur Absolue

Variables : N, V : Réel

Début

Ecrire("Une nombre : ")

Lire(N)

$V \leftarrow N$

Si $N < 0$ Alors

$V \leftarrow -N$

FinSi

Ecrire("La valeur absolue est : ", V)

Fin

Structures Conditionnelles Imbriquées

Exemple 4 :

Ecrire un algorithme qui détermine le signe d'un nombre entré par l'utilisateur.
On distingue deux cas : Strictement Positif ou Négatif

Algorithme Signe

Variables : N : Réel

Début

Ecrire("Une nombre : ")

Lire(N)

Si $N > 0$ Alors

Ecrire("Nombre strictement positif")

SiNon

Ecrire("Nombre négatif")

FinSi

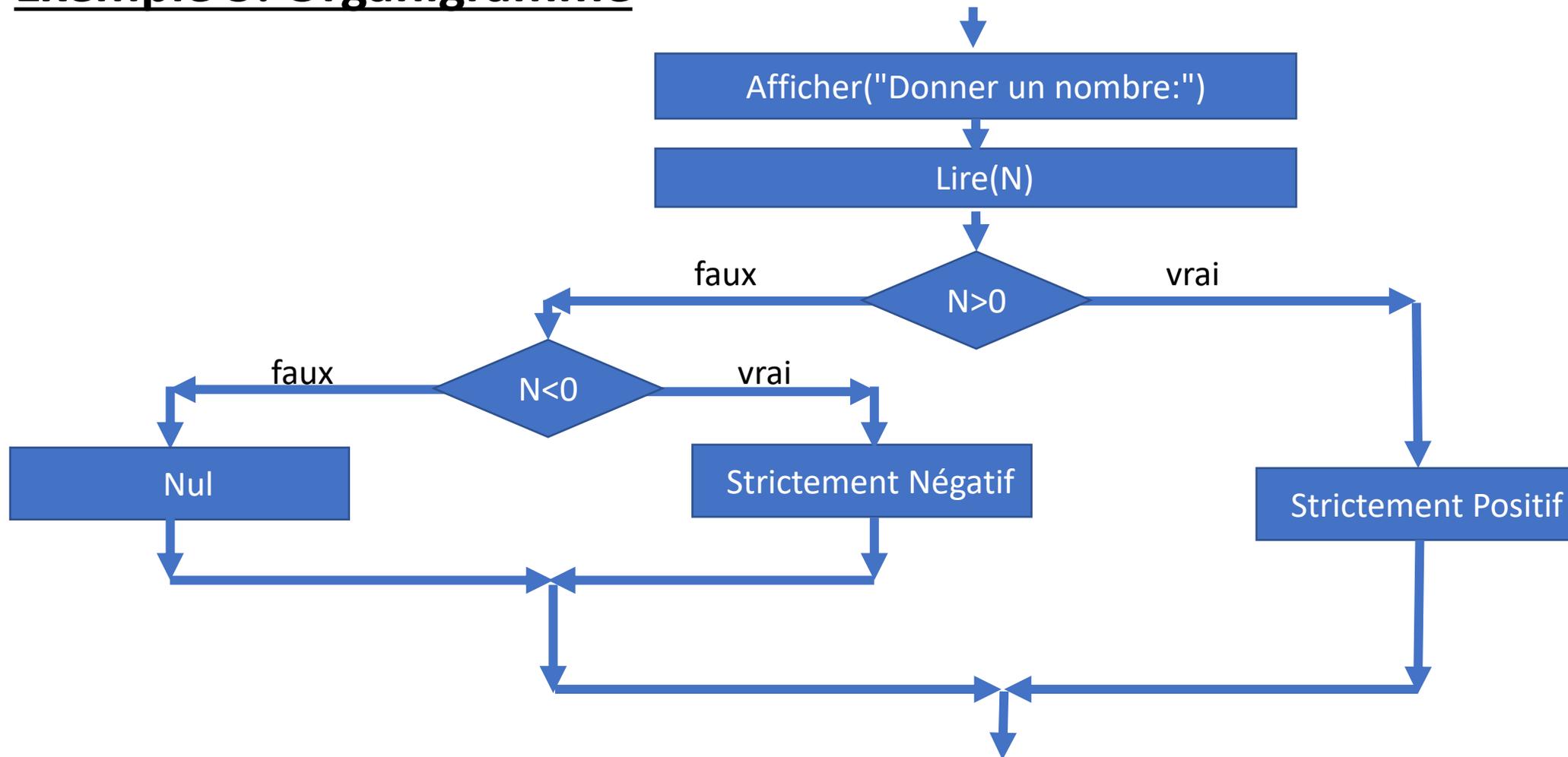
Fin

Exemple 5:

Améliorer l'algorithme du signe pour traiter les trois cas suivants :

- Strictement positif
- Strictement négatif
- nul

Exemple 5: Organigramme



Exemple 5: Algorithme en Pseudo-code

Algorithme Signe

Variables : N : Réel

Début

 Ecrire("Une nombre : ")

 Lire(N)

 Si $N > 0$ Alors

 Ecrire("Nombre strictement positif")

 SiNon

 Si $N < 0$ Alors

 Ecrire("Nombre négatif")

 SiNon

 Ecrire("Nul")

 FinSi

FinSi

Fin

Définition:

La structure conditionnelle multiple (Si...Sinon Si...Sinon / If...Elif...Else) est utilisée pour évaluer plusieurs conditions successives. Elle permet de tester plusieurs situations l'une après l'autre jusqu'à ce qu'une soit vraie. Si aucune des conditions n'est vraie, le bloc else est exécuté.

Syntaxe :

```
Si condition1 Alors  
    Traitement 1  
SiNon Si condition2 Alors  
    Traitement 2  
SiNon Si condition 3 Alors  
    Traitement 3  
...  
SiNon  
    Traitement par défaut  
FinSi
```

Structures conditionnelles

Structure conditionnelle multiple

Algorithme Signe

Variables : N : Réel

Début

 Ecrire("Une nombre : ")

 Lire(N)

 Si $N > 0$ Alors

 Ecrire("Nombre strictement positif")

 SiNon Si $N < 0$ Alors

 Ecrire("Nombre négatif")

 SiNon

 Ecrire("Nul")

 FinSi

Fin

Exemple 6 :

Écrivez un algorithme qui demande à l'utilisateur de saisir trois notes (entre 0 et 20) et qui calcule et affiche la moyenne de ces trois notes. Ensuite, le programme devrait afficher la mention associée à cette moyenne selon les critères suivants :

- Si la moyenne est inférieure à 10, affichez "Insuffisant".
- Si la moyenne est entre 10 inclus et 12 exclus, affichez "Passable".
- Si la moyenne est entre 12 inclus et 14 exclus, affichez "Assez bien".
- Si la moyenne est entre 14 inclus et 16 exclus, affichez "Bien".
- Si la moyenne est supérieure ou égale à 16, affichez "Très bien".

Structure conditionnelle multiple

Exemple 6 : Algorithme en pseudo-code

Algorithme Mention

Variables note1, note2, note3 : Réel

mention : chaîne

Début

Ecrire("Entrez la première note : ")

Lire(note1)

Ecrire("Entrez la deuxième note : ")

Lire(note2)

Ecrire("Entrez la troisième note : ")

Lire(note3)

moyenne = (note1 + note2 + note3) / 3

Fin

Si moyenne < 10 Alors

 mention ← "Insuffisant"

Sinon si moyenne < 12 Alors

 mention ← "Passable"

Sinon si moyenne < 14 Alors

 mention ← "Assez bien"

Sinon si moyenne < 16 Alors

 mention ← "Bien"

Sinon

 mention ← "Très bien"

FinSi

Ecrire("La moyenne est de ", moyenne)

Ecrire("la mention est : ", mention)

Exemple 7 :

Écrivez un algorithme résout les équations du premier degré de la forme :

$$A X + B = 0$$

- **Les donnée d'entré :**

Les coefficients A et B

- **Les données de sortie :**

La solution de l'équation X

- **Traitement :**

$$S = -B/A \quad \text{Si } A \neq 0$$

$$S = \mathbb{R} \quad \text{SI } A = 0 \text{ ET } B = 0$$

$$S = \emptyset \quad \text{SI } A = 0 \text{ ET } B \neq 0$$

Exemple 7 : Algorithme

Algorithme Equation

Variables A,B,S: Réel

Début

Ecrire("Donner A: ")

Lire(A)

Ecrire("Donner B: ")

Lire(B)

Si $A \neq 0$ Alors

$S \leftarrow -B/A$

 Ecrire("La solution est ", S)

Sinon si $B=0$ Alors

 Ecrire("Infinité de solutions")

Sinon

 Ecrire("Pas de solutions")

FinSi

Fin

Exercice :

Coder en Python les algorithmes conçus dans cette partie.

Exercice : Signe (2 cas)

```
N = float(input('Donner un nombre : '))
if N > 0 :
    print("Le nombre est strictement positif")
else :
    print('Le nombre est négatif')
```

Exercice : Signe (3 cas)

```
N = float(input('Donner un nombre : '))
if N > 0 :
    print("Le nombre est strictement positif")
elif N < 0 :
    print("Le nombre est strictement négatif")
else :
    print('Le nombre est nul')
```

Exercice : Equation premier degré

```
A = float(input('Donner le coefficient A : '))
B = float(input('Donner le coefficient B : '))
if A!=0 :
    S = -B/A
    print("La solution est : ", S)
elif B!=0 :
    print('Pas de solution')
else :
    print('Infinité de solutions')
```

Exercice : La mention

```
note1 = float(input("Entrez la première note : "))
note2 = float(input("Entrez la deuxième note : "))
note3 = float(input("Entrez la troisième note : "))
moyenne = (note1 + note2 + note3) / 3
if moyenne < 10 :
    mention = "Insuffisant"
elif moyenne < 12 :
    mention = "Passable"
elif moyenne < 14 :
    mention = "Assez bien"
elif moyenne < 16 :
    mention = "Bien"
else:
    mention = "Très bien"

print("La moyenne est de ", moyenne)
print("la mention est : ", mention)
```