



Ministère de l'enseignement supérieur et de la recherche scientifique
Université Djillali BOUNAAMA - Khemis Miliana (UDBKM)
Faculté des Sciences et de la Technologie
Département de Mathématiques et d'Informatique



Chapitre 3

Les Structures Conditionnelles

MI-L1-UEF121 : Algorithmiques et Structures de Données I

Nouredine AZZOUZA

n.azzouza@univ-dbkm.dz

Plan du Cours

1. Introduction

2. Structure conditionnelle simple : La conditionnelle

3. Structure conditionnelle composée : L'alternative

4. Structure conditionnelle de choix multiple

5. Le branchement

Introduction

Besoin

- ✓ Les problèmes sont plus complexes d'être résolus avec des instructions simples.
- ✓ La résolution de certains problèmes ne peut se faire que sous condition.
- ✓ Trouver une structure algorithmique capable de prendre en charge
 - les différents traitements relatifs aux différentes conditions
 - déclencher exclusivement le traitement qui respecte une certaine condition.



Bloc

- ✓ Un **bloc** est un ensemble cohérent de une ou plusieurs actions primitives
- ✓ Un **bloc** commence par un « **Début** » de bloc (Début, Dsi, Dpour, ...)
- ✓ Un **bloc** se termine par un « **Fin** » de bloc (Fin, Fsi, Fpour, ...)
- ✓ Si le **bloc** est composé d'une (1) seule actions alors, « **Début** » et « **Fin** » de bloc sont facultatifs



La Conditionnelle

La Conditionnelle

Syntaxe

```
Si condition Alors
```

```
    Dsi
```

```
        Bloc 1
```

```
    Fsi
```

```
...
```

- ✓ Si la **condition** est ***vérifiée*** (vraie), le « Bloc 1 » est exécuté.
- ✓ Si la **condition** n'est ***pas vérifiée*** (fausse), on passe en séquence après le « Bloc1 »
- ✓ La **condition** est une ***expression logique***

Exemple1 : Résolution d'équation de 1^{er} degré

Equation du 1^{er} degré

écrire un algorithme permettant de résoudre l'équation de 1^{er} degré : $ax+b=0$ (on suppose que $a > 0$)

Analyse :

Avec des notions en mathématiques,
La solution d'une équation de 1^{er} degré
est : $x = \frac{-b}{a}$

Algorithme equation_1er;

Var a, b, x: réel ;

Début

Lire (a, b);

Si (a <> 0) *Alors*
x ← -b/a

Fin

Ecrire ('la solution x = ', x);

Fin.

La Conditionnelle

```
float a, b, x;  
printf("Donner a et b : ");
```

PASCAL

C

Déclaration

Syntaxe: **IF** condition **THEN** Begin ... End

Syntaxe: **if** (condition) { ... }

Exemples

```
program Exemple_Const;  
  
var    a, b, x : Real;  
  
begin  
    Write('Donner a et b :');  
    ReadLn(a, b);  
  
    if (a <> 0) then  
        begin  
            x := -b/a;  
        end;  
  
    WriteLn('La solution x = ', x);  
  
end.
```

```
#include <stdio.h>  
  
int main (){  
  
    float a, b, x;  
    printf("Donner a et b : ");  
    scanf("%f %f", &a, &b);  
  
    if (a != 0)  
    {  
        x = -b / a;  
    }  
    printf("La solution x = %f", x);  
  
    return 0;  
}
```

L'Alternative

L'Alternative

Syntaxe

```
Si condition Alors  
    Dsi  
        Bloc 1  
    Fsi  
Sinon  
    Dsi  
        Bloc 2  
    Fsi  
...
```

- ✓ Si la **condition** est **vérifiée** (vraie), le « Bloc 1 » est exécuté.
- ✓ Si la **condition** n'est **pas vérifiée** (fausse), le « Bloc 2 » est exécuté.



Exemple1 : Positif? Négatif?

« Positif » ou « négatif »

Écrire un algorithme qui demande un entier à l'utilisateur, teste si ce nombre est positif ou non, et affiche « **positif** » ou « **négatif** ».

Analyse :

"Demande un entier de l'utilisateur " → Lire ...

"A est Positif" → $A \geq 0$

"A est négatif" → $A < 0$

Algorithme Pos_Neg;

Var A : entier ;

Début

Ecrire ('Donner un nombre A :');

Lire (A);

Si (A \geq 0) *Alors*

Ecrire (A, ' est positif');

Sinon

Ecrire (A, ' est négatif');

Fsi

Fin.

Exemple2 : Positif? Négatif? Nul?

« **Positif** », « **négatif** » ou « **nul** »

Écrire un algorithme qui demande un entier à l'utilisateur, et affiche « **strictement positif** », « **strictement négatif** » ou « **nul** ».

Analyse :

"A est strictement Positif" $\rightarrow A > 0$

"A est strictement négatif" $\rightarrow A < 0$

"A est nul" $\rightarrow A = 0$

Algorithme Pos_Neg_Nul;

Var A : entier ;

Début

Ecrire ('Donner un nombre A :');

Lire (A);

Si (A > 0) Alors

Ecrire (A, ' est strictement positif');

Sinon

Si (A < 0) Alors

Ecrire (A, ' est strictement négatif');

Sinon

Ecrire (A, ' est nul');

Fsi

Fsi

Fin.

Déclaration

PASCAL

Syntaxe: **IF** condition **THEN** Begin ... End
ELSE Begin ... End

Exemples

```
program Exemple_Const;

var
    A : Integer;

begin
    Write('Donner un nombre A :');
    ReadLn(A);

    if (A > 0) then
        WriteLn(A, ' est strictement positif')
    else
        if (A < 0) then
            WriteLn(A, ' est strictement négatif')
        else
            WriteLn(A, ' est nul');

end.
```

C

Syntaxe: **if** condition { ... } **else** { ... }

```
#include <stdio.h>

int main (){
    int a;
    printf("Donner un nombre A : ");
    scanf("%d", &a);

    if (a > 0){
        printf("%d est strictement positif", a);
    }else{
        if (a < 0){
            printf("%d est strictement négatif", a);
        }else{
            printf("%d est nul", a);
        }
    }
    return 0;
}
```

Conditionnelle

à Choix Multiple



Conditionnelle à Choix Multiple

Syntaxe

Selon *Variable ou Expression* **Faire**

Dselon

Valeur1 : *séquence 1*

Valeur2 : *séquence 2*

...

Valeurn : *séquence n*

Sinon *séquence par défaut*

Fselon

...

- ✓ Le **choix** se fait suivant la valeur d'un **sélecteur** (Variable ou expression)
- ✓ Cette structure permet d'éviter les **conditionnelles imbriqués**



Exemple : Menu de réservation

Menu de réservation

Analyse

Algorithme menu;

Var choix : entier ;

Debut

Ecrire('Entrer votre choix : ');

Ecrire('1. Reserver une vehicule');

Ecrire('2. Reserver une chambre');

Ecrire('3. Reserver un vol');

Lire (choix);

Selon choix Faire

1: Ecrire('1. Reserver une vehicule');

2: Ecrire('2. Reserver une chambre');

3: Ecrire('3. Reserver un vol');

sinon Ecrire('1. Reserver une vehicule');

Fselon

Fin.

Conditionnelle à Choix Multiple

PASCAL

Syntaxe: **CASE** variable **OF**

```
program Exemple_Const;

var
    choix : Integer;

begin
    WriteLn('Entrer votre choix : ');
    WriteLn('1. Réserver une véhicule');
    WriteLn('2. Réserver une chambre');
    WriteLn('3. Réserver un vol');

    ReadLn(choix);
    Case choix of:
        1: WriteLn('un véhicule est réservé');
        2: WriteLn('une chambre est réservé');
        3: WriteLn('un vol est réservé');
        else WriteLn('Choix invalide');
    end;
```

C

Syntaxe: **switch** (variable) { **case** ... }

```
#include <stdio.h>
int main (){
    int choix;
    printf("Entrer votre choix : \n");
    printf("1. Réserver une véhicule\n");
    printf("2. Réserver une chambre\n");
    printf("3. Réserver un vol\n");
    scanf("%d", &choix);
    switch (choix)
    {
        case 1: printf("un véhicule est réservé");
                break;
        case 2: printf("une chambre est réservée");
                break;
        case 3: printf("un vol est réservé");
                break;
        default: printf("Choix invalide");
                break;
    }
    return 0;
}
```



Ministère de l'enseignement supérieur et de la recherche scientifique
Université Djillali BOUNAAMA - Khemis Miliana (UDBKM)
Faculté des Sciences et de la Technologie
Département de Mathématiques et d'Informatique



Chapitre 3

Les Structures Conditionnelles

MI-L1-UEF121 : Algorithmiques et Structures de Données I

Nouredine AZZOUZA

n.azzouza@univ-dbkm.dz