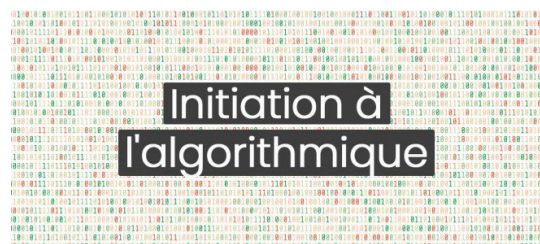


Chapitre 2 : Instructions conditionnelles

ALGORITHMIQUE (ALSD)



Préparé par Dr. BOUCHEBBAH Fatah

UNIVERSITÉ DE BÉJAIA

FACULTÉ DES SCIENCES EXACTES

DÉPARTEMENT D'INFORMATIQUE

Octobre 2022

Version 4

Table des matières



Objectifs	3
I - Introduction	4
II - Intérêt des instructions conditionnelles	5
III - Présentation d'une condition	6
IV - Instructions conditionnelles	7
1. Alternatives à un ou deux choix	7
1.1. <i>L'alternative simple</i>	7
1.2. <i>L'alternative complète</i>	8
2. Alternative à plusieurs choix	9
3. L'instruction SELON ... FAIRE (CASE ... OF)	10
V - Exercice	12
VI - Exercice	13
VII - Exercice	14
1. Exercice : Que doit afficher cet algorithme ?	15
2. Exercice : Que doit afficher cet algorithme ?	16
VIII - Test	17
Solutions des exercices	19
Références	22

Objectifs

A l'issue de ce chapitre, vous serez capable de :

- Définir une condition.
- Différencier entre les divers types d'alternatives.
- Employer efficacement les branchements conditionnels dans un algorithme.

Introduction



L'ordre dans lequel les instructions sont arrangées dans un algorithme est très important. Car, ces instructions sont effectuées une à une suivant l'ordre dans lequel elles y figurent. Cela veut dire, une instruction d'un algorithme ne peut être effectuée qu'après avoir effectué l'instruction qui lui est précédente, si elle existe. Mais parfois, certaines situations exigent qu'une ou plusieurs instructions soient négligées (ignorées ou sautées) dans une séquence d'instructions. Ce traitement est possible grâce aux *instructions conditionnelles*.

Ce chapitre porte sur les différentes instructions conditionnelles existantes. De ce fait, chaque instruction est décrite tout en donnant sa syntaxe, sa représentation en organigramme, et un exemple illustratif de son utilisation.

Intérêt des instructions conditionnelles



Pour illustrer l'importance des instructions conditionnelles, nous donnons l'algorithme ci-dessous qui calcule l'inverse d'un nombre :

```
1 ALGORITHME Inverse ;
2 VAR nombre : Réel ;
3 DÉBUT
4 ÉCRIRE ('Donner un nombre : ') ;
5 LIRE (nombre) ;
6 ÉCRIRE ('L'inverse du nombre : ', nombre, 'est : ', 1/nombre) ;
7 FIN.
```

Le programme associé à cet algorithme causera une fin anormale si le nombre tapé est nul (zéro). Dans cet exemple, l'ordinateur ne doit calculer l'inverse d'un nombre sauf si ce nombre n'est pas nul.

Les instructions de base vues jusqu'à présent : l'*affectation*, la *lecture* et l'*écriture* ne peuvent pas répondre à ce besoin, ce sont des instructions impératives.

Donc, nous avons besoins de nouvelles instructions, qui peuvent gérer de telles situations, qui sont *les instructions conditionnelles*.

Instructions conditionnelles

IV

Les instructions de contrôle utilisées en algorithmique se divisent en 2 grandes catégories : *les instructions conditionnelles* et *les instructions de bouclage*. Les instructions conditionnelles permettent de choisir entre des traitements différents, et les instructions de bouclage sont utilisées pour répéter plusieurs fois un même traitement. Dans cette section, nous détaillons l'ensemble des instructions conditionnelles existantes.


1. Alternatives à un ou deux choix

Ces instructions conditionnelles se basent sur la réalisation d'une seule condition pour choisir le traitement à exécuter. Elles peuvent prendre deux formes, à savoir : *l'alternative simple*, ou *l'alternative complète*.

1.1. L'alternative simple

L'alternative simple (appelée aussi *instruction conditionnelle SI ... ALORS*) est une instruction que nous utilisons lorsque nous voulons effectuer un traitement spécifique si une condition est vraie, et ne rien faire si la condition est fausse. Souvent la condition utilisée dans l'instruction SI ... ALORS est de comparaison.

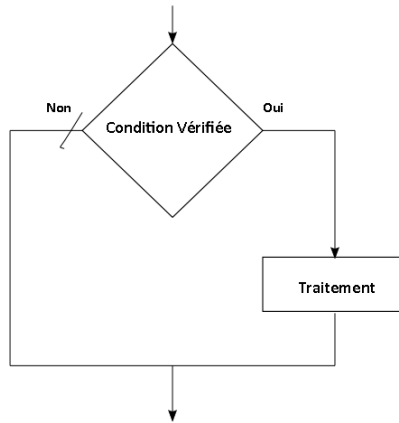
Syntaxe : L'instruction SI ... ALORS

La syntaxe algorithmique de cette instruction est ^{0,22}  :

```
1 SI (condition) ALORS
2 Traitement ;
3 FIN SI ;
```

En langage C :

```
1 if (condition)
2 {
3   Traitement ;
4 }
```



Organigramme de alternative simple.

👉 Exemple : La valeur absolue d'un entier

Écrire un algorithme qui calcule la valeur absolue d'un entier.

```

1 ALGORITHME valeur_absolue;
2 VAR Nombre : Entier ;
3 DÉBUT
4 LIRE (nombre) ;
5 SI (nombre < 0) ALORS
6   nombre <- -(nombre) ;
7 FIN SI;
8 ÉCRIRE ('La valeur absolue est', nombre) ;
9 FIN.
  
```

👉 Exemple : L'inverse d'un nombre : algorithme plus complet

Améliorer l'algorithme "Inverse" pour prendre en considération le cas de la division par zero.

```

1 ALGORITHME InverseComplet ;
2 VAR nombre: Réel ;
3 DÉBUT
4 ÉCRIRE ('Donner un nombre') ;
5 LIRE (nombre) ;
6 SI (nombre <> 0) ALORS
7   ÉCRIRE ('L'inverse du nombre : ', nombre, ' est : ', 1/nombre) ;
8 FIN Si ;
9 SI (nombre = 0) ALORS
10  ÉCRIRE ('le nombre tapé est nul') ;
11 FIN Si ;
12 FIN.
  
```

1.2. L'alternative complète

L'alternative complète (également appelée *instruction conditionnelle SI ... ALORS ... SINON*) exprime un choix entre deux traitements : *Traitement 1* et *Traitement 2*. Lorsque la condition de l'instruction est satisfaite (retourne VRAI), nous effectuons directement *Traitement 1*. Lorsque la condition n'est pas satisfaite (retourne FAUX), nous ignorons *Traitement 1* et nous effectuons *Traitement 2*.



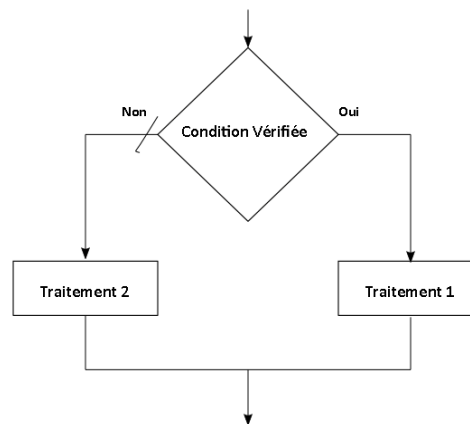
Syntaxe : L'instruction SI ... ALORS SINON

La syntaxe algorithmique de l'instruction est donnée comme suit^{p.22 ↗} :

```
1 SI (condition) ALORS
2 Traitement 1 ;
3 SINON
4 Traitement 2 ;
5 FIN SI ;
```

En langage C :

```
1 if (condition)
2 {
3 Traitement 1 ;
4 }else
5 {
6 Traitement 2 ;
7 }
```



Organigramme de l'alternative complète.



Exemple : L'inverse d'un nombre avec l'alternative complète

Écrire un algorithme qui calcule l'inverse d'un nombre en utilisant l'alternative complète.

```
1 ALGORITHME InverseAltcomp;
2 VAR nombre : Réel ;
3 DÉBUT
4 LIRE (nombre) ;
5 SI (nombre <> 0) ALORS
6 ÉCRIRE ('L'inverse du nombre : ', nombre, ' est : ', 1/nombre) ;
7 SINON
8 ÉCRIRE ('le nombre introduit est nul') ;
9 FIN SI ;
10 FIN.
```

2. Alternative à plusieurs choix

Dans le cas d'un choix multiple, il est possible d'imbriquer plusieurs instructions conditionnelles simples ou complètes pour former une structure plus complexe capable de supporter le nombre de

choix à faire.

Exemple : L'observation selon la moyenne d'un étudiant

Écrire un algorithme qui lit la moyenne d'un étudiant et lui affiche une observation. L'observation est :

- Admis si la moyenne est supérieure ou égale à 10.
- Racheté si la moyenne est strictement inférieure à 10 et supérieure ou égale à 9.5.
- Ajourné si la moyenne est strictement inférieure à 9.5.

```
1 ALGORITHME Observation ;
2 VAR Moyenne : Réel ;
3 DÉBUT
4 LIRE (Moyenne);
5 SI (Moyenne >= 10) ALORS
6 ÉCRIRE ('Admis');
7 SINON
8 SI (Moyenne >= 9.5) ALORS
9 ÉCRIRE ('Racheté');
10 SINON
11 ÉCRIRE ('Ajourné');
12 FIN SI ;
13 FIN SI ;
14 FIN.
```

3. L'instruction SELON ... FAIRE (CASE ... OF)

Cette instruction conditionnelle est appelée aussi *condition à choix sélectif*. Car, elle sélectionne entre plusieurs choix à la fois, et non entre deux choix alternatifs. La particularité de cette instruction est l'utilisation d'une expression arithmétique, contrairement aux instructions précédentes qui utilisent une ou plusieurs conditions.

Syntaxe : L'instruction SELON ... FAIRE

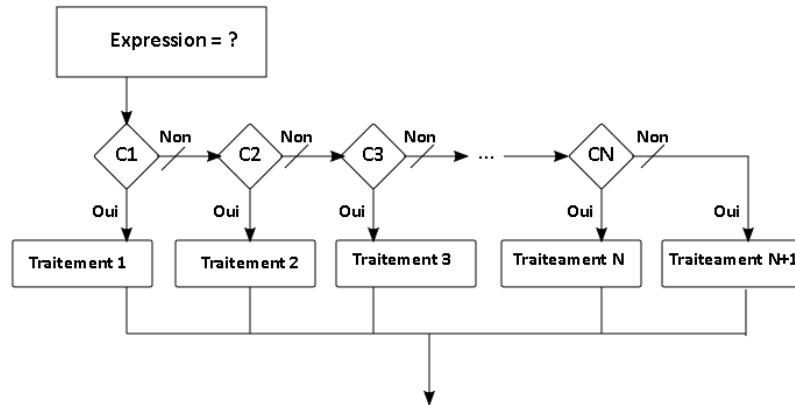
La syntaxe algorithmique de l'instruction SELON ... FAIRE est comme suit^{p.22 ↗} :

```
1 SELON (expression) FAIRE
2 CAS C1 : Traitement 1 ;
3 CAS C2 : Traitement 2 ;
4 CAS C3 : Traitement 3 ;
5 ...
6 CAS CN : Traitement N ;
7 CAS AUTRE Traitement (N+1) ;
8 FIN SELON ;
```

En langage C :

```
1 switch (expression)
2 {
3 case C1 : Traitement 1 ; break;
4 case C2 : Traitement 2 ; break;
5 case C3 : Traitement 3 ; break;
6 ...
7 case CN : Traitement N ; break;
```

```
8 default : Traitement (N+1) ;
9 END;
```



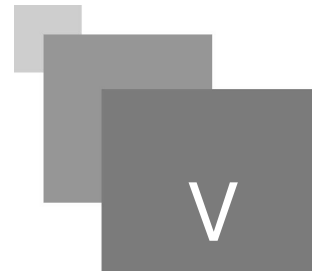
Organigramme de l'instruction conditionnelle SELON ... FAIRE.

Exemple : Le nombre de jours d'un mois

Écrire un algorithme qui affiche le nombre de jours d'un mois donné par l'utilisateur.

```
1 ALGORITHME jours_mois ;
2 VAR mois : Entier ;
3 DÉBUT
4 ÉCRIRE ('De quel mois s'agit-il?');
5 LIRE (mois) ;
6 SELON (mois) FAIRE
7 CAS 1 , 3 , 5 , 7 , 8 , 10 , 12 : ÉCRIRE ('Nombre de jours : 31' ) ;
8 CAS 4 , 6 , 9 , 11 : ÉCRIRE ('Nombre de jours : 30') ;
9 CAS 2 : ÉCRIRE ('Nombre de jours : 28 ou 29') ;
10 CAS AUTRE ÉCRIRE ('Ce mois n'existe pas') ;
11 FIN SELON ;
12 FIN.
```

Exercice



[solution n°1 p.19]

Ordonner les mots suivants pour reconstruire la syntaxe correcte d'une instruction conditionnelle.

FIN SI ;

SINON

Condition

SI

ALORS

Traitement 2 ;

Traitement 1 ;

Exercice

VI

[solution n°2 p.19]

Ordonner les mots suivants pour reconstruire la syntaxe correcte d'une instruction conditionnelle.

FAIRE

CAS 3 : Traitement 3 ;

FIN SELON ;

EXPRESSION

SELON

CAS 2 : Traitement 2 ;

CAS AUTRE Traitement 4 ;

CAS 1 : Traitement 1 ;

0. Exercice : Que doit afficher cet algorithme ?

```
ALGORITHME Alg1 ;  
  
VAR  
  
X, Y : Entier ;  
  
DÉBUT  
  
X <- 3 ;  
  
Y <- 2 ;  
  
SI (X<Y) ALORS  
  
ÉCRIRE (Y, ' est supérieur à', X) ;  
  
SINON  
  
SI (Y > 0) ALORS  
  
ÉCRIRE (Y, ' est strictement positif') ;  
  
SINON  
  
ÉCRIRE (Y, ' est inférieur ou égale à 0') ;  
  
FIN SI ;  
  
FIN SI ;  
  
FIN.
```

0. Exercice : Que doit afficher cet algorithme ?

```
ALGORITHME Alg2 ;
```

```
VAR
```

```
X, Y : Entier ;
```

```
DÉBUT
```

```
X <- 3 ;
```

```
Y <- 2 ;
```

```
SELON (X+Y-1) FAIRE
```

```
CAS 3 : ÉCRIRE ('Premier CAS') ;
```

```
CAS 5 : ÉCRIRE ('Deuxième CAS') ;
```

```
CAS 7 : ÉCRIRE ('Troisième CAS') ;
```

```
CAS AUTRE ÉCRIRE ('CAS AUTRE') ;
```

```
FIN SELON ;
```


Test



Exercice

[solution n°4 p.20]

L'instruction conditionnelle simple est basée sur une seule condition.

- Vrai
- Faux

Exercice

[solution n°5 p.20]

Dans l'instruction conditionnelle simple, le traitement qui est à l'intérieur de la condition est effectué si la condition n'est pas vérifiée.

- Vrai
- Faux

Exercice

[solution n°6 p.21]

L'instruction conditionnelle complète impose un choix entre deux traitements différents en se basant sur trois conditions.

- Vrai
- Faux



Exercice

[solution n°7 p.21]

Un étudiant a écrit la condition $3 < X < 6$. Son enseignant lui a dit que cette condition est mal formée.

Selon vous, quelle est (sont) la (les) forme (s) correcte (s) de la condition parmi les conditions ci-dessous :

- $(3 < X)$ ou $X < 6$
- $3 < X$ et $X < 6$
- $(3 < X)$ ou $(X < 6)$
- $(3 < X)$ et $(X < 6)$
- $(X > 3)$ et $(X < 6)$

Exercice

[solution n°8 p.21]

Quelle est (sont) la (les) affirmation (s) correcte (s) concernant l'instruction conditionnelle SELON ... FAIRE parmi les conditions ci-dessous :

- Elle est basée sur une condition.
- Elle est basée sur une expression.
- Elle sélectionne entre plusieurs choix à la fois.
- Elle sélectionne entre deux choix alternatifs.
- Elle peut être imbriquée dans une instruction conditionnelle simple ou complète.

Solutions des exercices

> Solution n° 1

Exercice p. 12

Ordonner les mots suivants pour reconstruire la syntaxe correcte d'une instruction conditionnelle.

SI Condition ALORS Traitement 1 ; SINON Traitement 2 ; FIN SI ;

> Solution n° 2

Exercice p. 13

Ordonner les mots suivants pour reconstruire la syntaxe correcte d'une instruction conditionnelle.

SELON EXPRESSION FAIRE CAS 1 : Traitement 1 ; CAS 2 : Traitement 2 ;
CAS 3 : Traitement 3 ; CAS AUTRE Traitement 4 ; FIN SELON ;

> Solution n° 3

Exercice p. 14

Exercice : Que doit afficher cet algorithme ?

```
ALGORITHME Alg1 ;
VAR
X, Y : Entier ;
DÉBUT
X <- 3 ;
Y <- 2 ;
SI (X<Y) ALORS
ÉCRIRE (Y, ' est supérieur à', X) ;
SINON
SI (Y > 0) ALORS
```


- Vrai
- Faux

> **Solution n° 6**

Exercice p. 17

L'instruction conditionnelle complète impose un choix entre deux traitements différents en se basant sur trois conditions.

- Vrai
- Faux

> **Solution n° 7**

Exercice p. 18

Un étudiant a écrit la condition $3 < X < 6$. Son enseignant lui a dit que cette condition est mal formée.

Selon vous, quelle est (sont) la (les) forme (s) correcte (s) de la condition parmi les conditions ci-dessous :

- $(3 < X)$ ou < 6
- $3 < X$ et $X < 6$
- $(3 < X)$ ou $(X < 6)$
- $(3 < X)$ et $(X < 6)$
- $(X > 3)$ et $(X < 6)$

> **Solution n° 8**

Exercice p. 18

Quelle est (sont) la (les) affirmation (s) correcte (s) concernant l'instruction conditionnelle SELON ... FAIRE parmi les conditions ci-dessous :

- Elle est basée sur une condition.
- Elle est basée sur une expression.
- Elle sélectionne entre plusieurs choix à la fois.
- Elle sélectionne entre deux choix alternatifs.
- Elle peut être imbriquée dans une instruction conditionnelle simple ou complète.

Références



1

L. Baba-Hamed, S. Hocine. Algorithmique et structures de données : Cours et exercices avec solutions. Office des Publications Universitaires, 2006

2

M. C. Belaid. Algorithmique et programmation en Pascal, cours, exercices et travaux pratique avec corrigés. Edition Les Pages Bleues, 2004.

3

M. C. Belaid. Algorithme et programmation en Pascal. Edition les pages bleus, 2006.

4

T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. Algorithmique : Cours avec 957 exercices et 158 problèmes. Édition DUNOD, 3ème édition, 2010.

5

J. Courtin. Initiation à l'algorithmique et aux structures de données. Edition DUNOD, 1998.

6

M. Divay. Algorithmes et structures de données génériques. Edition Dunod, 2004.

7

L. Goldschlager and A. Lister. Informatique et algorithmique. InterEditions. 1986.

8

<https://9alami.info/cours-informatique/liste-des-modules/lecon1-notion-dalgorithme> (Consulté le 09/03 /2021)

9

<https://www.youtube.com/playlist?list=PL2aehqZh72Lumvy4tSekr6Rzcgwn15MLI> (Consulté le 05/04 /2021)