

CHAPITRE3

LES STRUCTURES CONDITIONNELLES

1.Introduction :

Souvent les problèmes nécessitent l'étude de plusieurs situations qui ne peuvent pas être traitées par les séquences d'actions simples. Puisqu'on a plusieurs situations, et qu'avant l'exécution, on ne sait pas à quel cas de figure on aura à exécuter, dans l'algorithme on doit prévoir tous les cas possibles. Ce sont les structures conditionnelles qui le permettent, en se basant sur ce qu'on appelle condition.

Ils existent deux type de structures conditionnelles celle ou on a le choix entre deux traitement possibles on les appelle structures alternatives et celle ou on a le choix entre plusieurs traitements possibles on les appelle structures de choix.

2. Structure Alternative :

Cette structure n'offre que deux issues possibles à la poursuite de l'algorithme et s'excluant mutuellement.

2.1 Alternative simple:

Dans cette structure l'exécution de la séquence d'instruction n'est faite que si et seulement si la condition est vérifiée.

Syntaxe

Si (condition vérifiée) **alors**

Instruction1

Instruction 2

...

Fin si

L'exécution de cette structure se fait selon l'organigramme suivant :

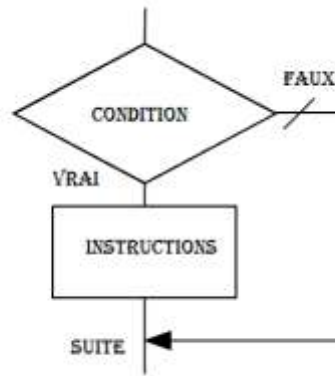


Figure 5- L'organigramme de l'exécution de la structure alternative simple

Exercice d'application 1 :

Ecrire un algorithme qui permet de saisir un entier et de le multiplier par deux s'il est supérieur à 10 et de l'afficher après.

Correction :

Algorithme Multiplication

Variables

X : entier

Début

Ecrire ("donner un entier")

Lire (X)

Si (X > 10) alors

X ← X*2

Ecrire(X)

Fin Si

Fin

Exercice d'application 2 :

Ecrire un algorithme qui permet de calculer le salaire d'un employeur en fonction du nombre d'heures qu'il a travaillé, du tarif de l'heure et du nombre d'année de service. Si le nombre d'année est supérieur à 10 ans on lui ajoute à son salaire de base une prime de 100 dinars.

Correction :

```
Algorithme Salaire

Variables

annee, nbh, tarif, salaire : réel

Début

    Ecrire ("donner le nombre d'heures travaillées")

    Lire(nbh)

    Ecrire("donner le tarif de l'heure")

    Lire(tarif)

    Ecrire("donner le nombre d'années travaillées ")

    Lire(annee)

    salaire ← nbh*tarif

    Si (annee>10) alors

        salaire ← salaire +100

    Fin Si

    Ecrire("le salaire est ",salaire)

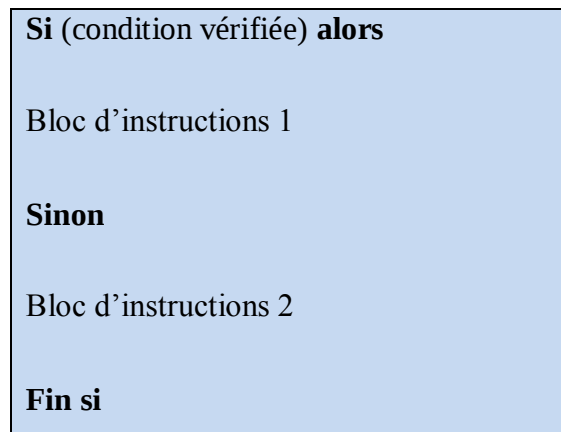
Fin
```

2.2 Alternative complète:

Dans cette structure l'exécution d'un des deux traitements distincts ne dépend que du résultat d'un test effectué sur la condition qui peut être une variable ou un événement :

- Si la condition est vérifiée seul le 1^{er} bloc d'instructions est exécuté.
- Si la condition est non vérifiée seul le 2^{ème} bloc d'instructions est exécuté.

Syntaxe :



Voici l'organigramme qui décrit l'exécution de cette structure :

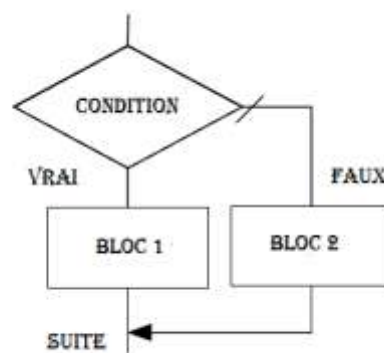
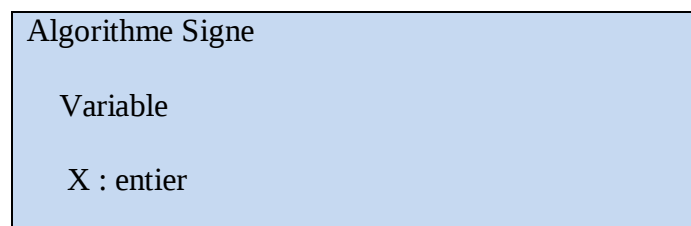


Figure 6- L'organigramme de l'exécution de la structure alternative complète

Exercice d'application 3 :

Ecrire un algorithme qui permet de saisir un entier et d'afficher s'il est positif ou négatif.

Correction :



```
Début

Ecrire("donner un entier ")

Lire(X)

Si (X >=0) alors

    Ecrire("l'entier est positif")

Sinon

    Ecrire("l'entier est négatif")

Fin si

Fin
```

Exercice d'application 4 :

Ecrire un algorithme qui permet de saisir un entier et d'afficher s'il est pair ou impair

Correction :

```
Algorithme Parité

Variable

X : entier

Début

    Ecrire("donner un entier ")

    Lire(X)

    Si (X Mod 2 =0) alors

        Ecrire("l'entier est pair")

    Sinon

        Ecrire("l'entier est impair")

    Fin si

Fin
```

3. Structure de choix :

La structure de choix permet, en fonction de plusieurs conditions de type booléen, d'effectuer des actions différentes suivant les valeurs que peut prendre une même variable.

Syntaxe

```
Selon 'sélecteur' faire  
  
Liste valeurs 1 : bloc d'instructions 1  
  
Liste valeurs 2 : bloc d'instructions 2  
  
...  
  
Liste valeurs n : bloc d'instructions n  
  
Sinon  
  
Autre bloc d'instructions  
  
Fin selon
```

Le $i^{\text{ème}}$ bloc d'instructions est exécuté si la valeur du sélecteur est dans l'intervalle des $i^{\text{ème}}$ valeurs.

L'organigramme de l'exécution de cette structure est le suivant :

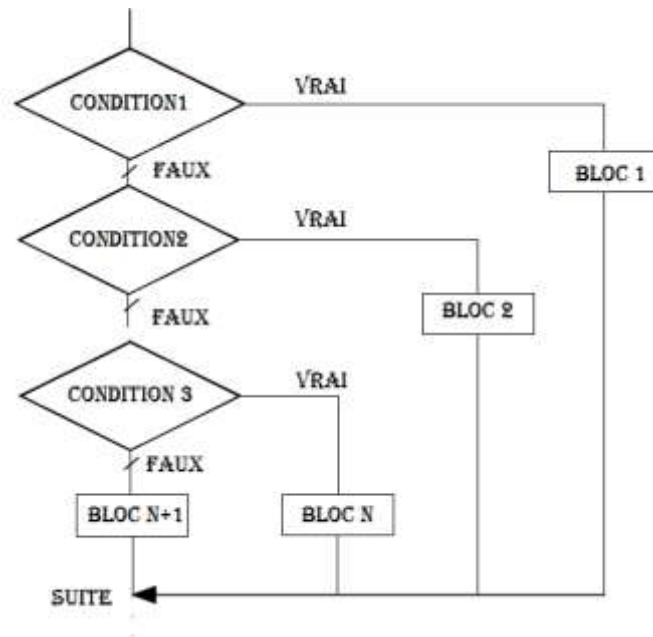


Figure 7- L'organigramme de l'exécution de la structure de choix

Remarque :

Il existe une alternative à cette structure mais qui n'est pas très conseillée de l'utiliser. Cette alternative consiste à l'utilisation des Si « imbriquées ».

Syntaxe :

```

Si (condition 1) alors
  Bloc 1
Sinon
  Si (condition 2) alors
    Bloc 2
  Sinon
    Si (condition 3) alors
      ....
    Fin si
  Fin si
Fin si
  
```

Exercice d'application 5

Ecrire un algorithme qui permet de saisir un entier entre 1 et 7 et d'afficher le jour correspondant.

Correction

```
Algorithme Jour

Variable

X : entier

Début

    Ecrire("donner un entier ")

    Lire(X)

    Selon X faire

        1 : Ecrire("Lundi")

        2 : Ecrire("Mardi")

        3 : Ecrire("Mercredi")

        4 : Ecrire("Jeudi")

        5 : Ecrire("Vendredi")

        6 : Ecrire("Samedi")

        7 : Ecrire("Dimanche")

    Sinon

        Ecrire("valeur incorrecte")

    Fin selon

Fin
```

Exercice d'application 6

Refaire l'exercice précédent en utilisant des Si « imbriquées »