

Chapitre n°3 :

Les structures de contrôle conditionnelles

Objectifs du cours :

- Maîtriser la structure Si.
- Maîtriser la structure Selon.

**Leçon 1****La structure de contrôle conditionnelle simple**

## I - La forme réduite

1. **Syntaxe et vocabulaire** : (voir livre page 72, 73)
2. **Définition** : (voir livre page 75)
3. **Remarques**
  - L'initialisation est le fait d'affecter une valeur initiale à un objet.
  - En Pascal, si le traitement après "Then" est composé d'une seule instruction, cette dernière peut être présentée sans les deux délimiteurs Begin et End

**Activité 1** :

Écrire une analyse qui permet de saisir un caractère et une chaîne puis afficher un message pour dire si le caractère existe dans la chaîne ou non.

## a. Analyse

Nom : <b>recherche</b>		
S	L.D.E	O.U
4	Résultat = Ecrire (c, msg, ch)	c
3	msg = [msg ← " N'existe pas "] Si POS (c, ch) ≠ 0 alors msg ← " Existe " Fin si	ch msg
1	c = Donnée ("Donner un caractère")	
2	ch = Donnée ("Donner une chaîne")	
5	Fin <b>recherche</b>	

**T.D.O**

Objet	Type	Rôle
ch	Chaîne	
c	Caractère	Caractère recherché
msg	Chaîne	message

**b. Algorithme**

- 0) **Début** recherche
- 1) écrire ("Donner un caractère")  
Lire (c)
- 2) écrire ("Donner une chaîne")  
Lire (ch)
- 3) msg ← " N'existe pas "  
Si POS (c, ch) <> 0 alors msg ← " Existe "  
Fin si
- 4) Ecrire (c, msg, ch)
- 5) **Fin** recherche

**c. Traduction en Pascal** (voir fichier : if\_pos.pas)**II - La forme alternative**

1. **Syntaxe et vocabulaire** : (voir livre page 77, 78)
2. **Définition** : (voir livre page 80)

**Activité 2** :

Saisir une chaîne de caractère et vérifier si elle est composée de plusieurs mots.

**a. Analyse**

Nom : <b>phrase</b>		
S	L.D.E	O.U
	Résultat = Affichage	
2	Affichage = [ ] Si POS (" ", ch) = 0 alors écrire (" Mot ") Sinon écrire (" Phrase ") Fin si	ch
1	ch = Donnée ("Donner une chaîne : ")	
3	Fin <b>phrase</b>	

**T.D.O**

Objet	Type	Rôle
ch	Chaîne	

**b. Algorithme**0) **Début**

1) écrire ("Donner une chaîne")

Lire (ch)

2) Si POS (" ", ch) = 0 alors écrire ("Votre chaîne est composée d'un seul mot ")

Sinon écrire ("Votre chaîne est composée de plusieurs mots ")

Fin si

3) **Fin****c. Traduction en Pascal (voir fichier : phrase1.pas)**

## • Cas particulier :

ch contient "" message  $\otimes$  "Votre chaîne est composée d'un seul mot" **Faux !**ch contient " o " message  $\otimes$  "Votre chaîne est composée de plusieurs mots" **Faux !**ch contient " " message  $\otimes$  "Votre chaîne est composée de plusieurs mots" **Faux !****Version 2**0) **Début**

1) écrire ("Donner une chaîne"), Lire (ch)

2) Tant que (ch [1] = " ") et (long (ch)  $\neq$  0) faire

Efface (ch, 1, 1)

Fin Tant que

3) Tant que ch [Long (ch)] = " " et (long (ch)  $\neq$  0) faire

Efface (ch, Long (ch), 1)

Fin Tant que

4) **Si** Long (ch) = 0 **alors** écrire ("La chaîne est vide ")**Sinon Si** POS (" ", ch) = 0 **alors** écrire ("Votre chaîne est composée d'un seul mot")**Sinon** écrire ("Votre chaîne est composée de plusieurs mots ")**Fin si**5) **Fin****d. Traduction en Pascal (voir fichier : phrase2.pas)**

## Leçon 2

## La structure de contrôle conditionnelle généralisée

I - Vocabulaire et syntaxe (voir livre page 86, 87)

II - Définition (voir livre page 90)

### Activité 1 : ⌚

Écrire un programme qui affiche l'heure système, Avancer cet horaire d'une seconde et l'afficher au milieu de la fenêtre d'exécution.

Exemple : 15:34:26      ↘      15:34:27

### a. Analyse

S	L.D.E	O.U
4	Résultat = Écrire (h, ":", m, ":", s)	h
3	h, m, s = [ ]	m
	Si s < 59 alors s ← s + 1	s
	Sinon Si m < 59 alors	
	s ← 0	
	m ← m + 1	
	Sinon Si h < 23 alors	
	s ← 0	
	m ← 0	
	h ← h + 1	
	Sinon	
	s ← 0	
	m ← 0	
	h ← 0	
	Fin si	
2	Écrire (h, ":", m, ":", s)	
1	Gettime (h, m, s, c)	c
5	Fin <b>heure_système</b>	

### T.D.O

Objet	Type	Rôle
h	mot	Heure
m	mot	Minute
S	mot	Seconde
c	mot	Centième de seconde

## b. Algorithme

- 0) **Début** heure\_système
- 1) Gettime (h, m, s, c)
- 2) Écrire (h, ":", m, ":", s)
- 3) Si  $s < 59$  alors  $s \leftarrow s + 1$
- Sinon Si  $m < 59$  alors
- $s \leftarrow 0$
- $m \leftarrow m + 1$
- Sinon Si  $h < 23$  alors
- $s \leftarrow 0$
- $m \leftarrow 0$
- $h \leftarrow h + 1$
- Sinon
- $s \leftarrow 0$
- $m \leftarrow 0$
- $h \leftarrow 0$
- Fin si
- 4) Écrire (h, ":", m, ":", s)
- 5) **Fin** heure\_système

## c. Traduction en Pascal (voir fichier : heure\_sy.pas)

**Activité 2 :** Écrire une analyse, un algorithme et la traduction en pascal du programme intitulé équation1, qui détermine est affiche les solutions d'une équation du 1<sup>er</sup> ordre de la forme  $ax + b = 0$ . Avec  $a$  et  $b \in \mathbb{R}$

## a. Pré analyse :

*Résultat :* Afficher  $x$

*Traitement :* 1<sup>er</sup> cas  $a \neq 0$  alors  $x = -\frac{b}{a}$

2<sup>ème</sup> cas  $a = 0$  alors 1)  $b = 0$  alors écrire IR

2)  $b \neq 0$  alors écrire { }

*Données :*  $a$  et  $b$  entier.

## b. Analyse :

Nom : équation1		
S	L.D.E	O.U
	<b>Résultat</b> = R	
3	R = [ ] <b>Si</b> $a \neq 0$ <b>alors</b> écrire (-b/a) <b>Sinon si</b> $b = 0$ <b>alors</b> écrire ("IR") <b>Sinon</b> écrire ("{}")	a b
	<b>Fin si</b>	
2	a = Donnée ("Donner a : ")	
1	b = Donnée ("Donner b : ")	
4	<b>Fin</b> équation1	

Objet	Type	Rôle
a , b	Réel	

## c. Algorithmme

- 0) **Début** équation1
- 1) écrire ("Donner a : ") , lire (a)
- 2) écrire ("Donner b : ") , lire (b)
- 3) Si  $a \neq 0$  alors écrire (-b/a)  
    Sinon Si  $b = 0$  alors écrire ("IR")  
        Sinon écrire ("{}")  
    Fin si
- 4) **Fin** équation1

## d. Traduction Pascal

```

Program equation1;
uses wincrt;
var
  a,b : real;
begin
  write ('Donner a : ');
  readln (a);
  write ('Donner b : ');
  readln (b);

  if a<>0 then writeln(-b/a:5:2)
    else if b = 0 then writeln('x appartient à IR')
      else writeln('la solution est l"ensemble vide');

End.
```

## La structure de contrôle conditionnelle à choix

### I - Introduction

#### Activité 1 :

Ecrire un programme qui saisit le numéro d'un mois et affiche la saison correspondante.

Exemple : si numéro du mois = 8, le programme affichera : saison été.

#### a. Pré analyse

Résultat :

- Afficher un message qui indique le nom de la saison du mois

Traitements :

- mois  $\in$  {12, 1, 2}  $\Leftrightarrow$  afficher "Saison Hiver"
- mois  $\in$  {3, 4, 5}  $\Leftrightarrow$  afficher "Saison Printemps"
- mois  $\in$  {6, 7, 8}  $\Leftrightarrow$  afficher "Saison Eté"
- mois  $\in$  {9, 10, 11}  $\Leftrightarrow$  afficher "Saison Automne"

Donnée :

- mois

#### b. Analyse

Nom : Saison		
S	L.D.E	O.U
2	<p><b>Résultat</b> = Affichage</p> <p>Affichage = [ ] Selon <b>mois</b> Faire</p> <p style="padding-left: 40px;">12, 1, 2 : écrire ("Saison Hiver")</p> <p style="padding-left: 40px;">3.. 5 : écrire ("Saison Printemps")</p> <p style="padding-left: 40px;">6 .. 8 : écrire ("Saison Eté")</p> <p style="padding-left: 40px;">9.. 11 : écrire ("Saison Automne")</p> <p style="padding-left: 80px;">Sinon</p> <p style="padding-left: 80px;">Ecrire ("Erreur !")</p> <p style="padding-left: 40px;">Fin Selon</p>	mois
1	mois = Donnée ("saisir le n° d'un mois : ")	
3	<b>Fin</b> Saison	

T. D. O

Objets	Type	Rôle
mois	Entier	

### c. Algorithme

#### 0) Début Saison

1) Ecrire ("saisir votre Mois"), lire (**mois**)

2) Selon **mois** Faire

12, 1, 2 : écrire ("Saison Hiver")

3 .. 5 : écrire ("Saison Printemps")

6 .. 8 : écrire ("Saison Eté")

9 .. 11 : écrire ("Saison Automne")

Sinon

Ecrire ("Erreur !")

Fin Selon

3) **Fin** Saison

#### d. Traduction pascal (voir fichier : saison.pas)

## II - Syntaxe et commentaires (voir livre page 96)

## III - Définition (voir livre page 97)

Au niveau de l'analyse et de l'algorithme	Au niveau de turbo pascal
<pre>[ ] <b>Selon</b> sélecteur <b>faire</b>   Valeur 1 : Action1   Valeur 2 : Action2-1                 Action2-2                 .....                 Action2-n    Valeur 3, Valeur 4, Valeur 5 : Action3   Valeur 6 .. Valeur 7 : Action4   .....   Valeur n : Action n <b>Sinon</b>   Action R <b>Fin Selon</b></pre>	<pre>{Initialisation}; <b>Case</b> sélecteur <b>OF</b>   Valeur 1 : Action1;   Valeur 2 : <b>Begin</b>                 Action2-1 ;                 Action2-2 ;                 ..... ;                 Action2-n ;                 <b>End ;</b>    Valeur 3, Valeur 4, Valeur 5 : Action3 ;   Valeur 6 .. Valeur 7 : Action4 ;   ..... ;   Valeur n : Action n ;  <b>Else</b>   Action R ;  <b>End ;</b></pre>