

Chapitre 3

Les structures de contrôle conditionnelles

Leçon 1 :

Les structures de contrôle conditionnelles simples

I-Définition :

* La structure de contrôle conditionnelle permet à un programme de modifier son traitement en fonction d'une condition.

* Il existe trois formes d'instructions conditionnelles :

- ✓ Forme simple
- ✓ Forme généralisée.
- ✓ Forme à choix

II-La structure de contrôle conditionnelle simple :

1- A-La forme réduite :

1-Définition:

Une structure de contrôle conditionnelle est dite à forme simple réduite lorsque le traitement dépend d'une condition. Si la condition est évaluée à « vrai », le traitement est exécuté.

2-Vocabulaire et syntaxe:

Analyse	Algorithme	Pascal
[Init] Si condition Alors Instruction 1 Instruction 2 Instruction N FinSi {Init} Si condition Alors Instruction 1 Instruction 2 Instruction N FinSi ; {Init} IF condition THEN Begin Instruction_1; Instruction_2;; Instruction_N; End;;

Remarque : [Init] est une séquence d'instructions qui contiendra les éventuelles initialisations

Activité 1 : Écrire un algorithme qui permet de faire la racine carrée d'un réel x.

- 0) **Début** racine
- 1) Ecrire ("Donner un réel ") , lire(x)
- 2) **Si** $x \geq 0$ **alors** écrire("La racine carrée est =" , racine_carrée(x))
FinSi
- 3) **Fin** racine

Activité2 :

Écrire un programme qui permet de vérifier la parité d'un entier donné (en utilisant la forme simple réduite)

Algorithme :

- 0) **Début** parité
- 1) Ecrire("Donner un entier "), lire (n)
- 2) message ← "Pair"
- 3) **Si** $n \text{ mod } 2 \neq 0$ **alors**
 message ← "impair"

FinSi

- 4) Ecrire(n, " est ",message)
- 5) **Fin** parité

T.D.O

Objet	Type/Nature	Rôle
n	Entier	Entier à vérifier(parité)
message	chaîne	Contient le message à afficher

En Pascal

```

program parite ;
uses wincrt ;
var n:integer ;
    message : string ;
begin
writeln('Donner un entier = ') ;
readln(n) ;
message:='pair' ;
if n mod 2<>0 then
                message:='impair';
writeln(n, ' est ', message) ;
end.
    
```

2- La forme alternative :

1-Définition :

Une structure de contrôle conditionnelle est dite à forme alternative lorsque le traitement dépend d'une condition à deux états : Si la condition est évaluée à « vrai », le premier traitement est exécuté ; Si la condition est évaluée à « faux », le second traitement est exécuté.

2-Vocabulaire et syntaxe:

Analyse	Algorithme	Pascal
[Init] Si condition Alors Instruction 1 de TR1 Instruction 2 de TR1 Instruction m de TR1 Sinon Instruction 1 de TR2 Instruction 2 de TR2 Instruction n de TR2 FinSi {Init} Si condition Alors Instruction 1 de TR1 Instruction 2 de TR1 Instruction m de TR1 Sinon Instruction 1 de TR2 Instruction 2 de TR2 Instruction n de TR2 FinSi ; {Init} IF condition THEN Begin Instruction_1_de_TR1; Instruction_2_de_TR1;; Instruction_m_de_TR1; End ELSE Begin Instruction_1_de_TR2; Instruction_2_de_TR2;; Instruction_n_de_TR2; End;

Remarque :

On ne met pas « ; » après le END qui précède le ELSE car la structure conditionnelle n'est pas encore terminée.

Activité : Ecrire un programme qui permet de vérifier la parité d'un entier (utilisant la forme alternative)

Algorithme :

- 0) Début parité2
- 1) Ecrire("Donner un entier "), lire (n)
- 2) si $n \bmod 2 = 0$ alors écrire("Pair")
- 3) sinon écrire("impair")
 - a. FinSi
- 4) Fin parité

T.D.O

Objet	Type/Nature	Rôle
n	Entier	Entier à vérifier(parité)

en pascal

```

program parite2 ;
uses wincrt ;
var
n:integer ;
begin
writeln('Donner un entier ') ;
readln (n) ;
if n mod 2 = 0 then
    writeln('Pair')
else
    writeln('impair') ;
End.
```

{on ne met pas un ;avant le else de if}

Remarque : On peut faire un affichage mixte : `writeln(n, 'est impair') ;`

Leçon 2 :

La structure de contrôle conditionnelle généralisée

I-Définition :

Une structure de contrôle conditionnelle est dite généralisée lorsqu'elle permet de résoudre des problèmes comportant plus de deux traitements en fonction des conditions. L'exécution d'un traitement entraîne automatiquement la non-exécution des autres traitements.

2-Vocabulaire et syntaxe :

Analyse & Algorithme	Pascal
[Init] Si condition 1 Alors Traitement 1 Sinon Si condition 2 Alors traitement 2 Sinon Si condition 3 Alors traitement 3 Sinon Si condition N-1 Alors traitement N-1 Sinon traitement N FinSi ; {Init} IF condition_1 THEN Traitement_1 ELSE IF condition_2 THEN traitement_2 ELSE IF condition_3 THEN traitement_3 ELSE IF condition_N-1 THEN traitement_N-1 ELSE traitement_N, ;

Remarque :

- ✓ Il est préférable de mettre les événements les plus probables en premier lieu.
- ✓ Chaque traitement peut comporter une ou plusieurs instructions.

Activité :

Écrire un programme qui permet de faire la résolution de l'équation du 1er degré : $ax+b = 0$

1) Algorithme

- 0) **Début** Equation
- 1) Ecrire (" Donner a : "), lire(a)
- 2) Ecrire ("Donner b : "), lire(b)
- 3) **si** $a \neq 0$ **alors** $S \leftarrow -b/a$
 |
 | Ecrire ("La solution est ", S)
 | **sinon**
 | | **si** $b=0$ **alors** Ecrire("La solution est IR ")
 | | **Sinon** Ecrire("La solution est vide ")
 | | Finsi
 | **FinSi**
- 4) **Fin** Equation

T.D.O

Objet	Type/nature	Rôle
a,b	Réel	Paramètres à saisir
S	réel	Une Solution de l'équation à calculer

2)Programme Pascal :

```

program equation ;
uses wincrt ;
var
    a,b,s : real ;
begin
    writeln('Donner a : ') ;      readln(a) ;
    writeln('Donner b : ') ;      readln(b) ;
    if (a<>0) then
        begin
            s := -b/a ;
            writeln('La solution est', s:5:2) ;
        End
    else
        if (b=0) then
            writeln('La solution est IR')
        else
            writeln('La solution est vide') ;
    end.

```

Exercice :

Écrire un programme qui permet de saisir une moyenne (moy) puis affiche la décision correspondante :

- ADMIS Si moy >=10
- CONTROLE Si 9<=moy <10
- REDOUBLE Si moy<9

Corrigé :

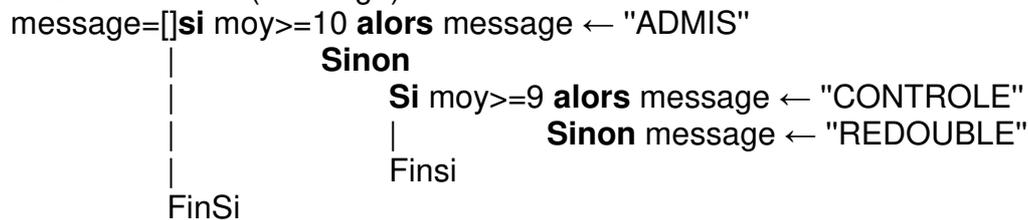
Pré-analyse

But : afficher la décision
 données : moy
 formules : structure si sur moy

Analyse :

Nom: Décision

Résultat= Ecrire(message)



moy=Donnée("Donner une moyenne ")

Fin Décision

T.D.O

Objet	Type/nature
Moy	Réel
Message	Chaîne de caractère

Algorithme :

- 0) Début décision
- 1) Écrire("Donner une moyenne") , lire(Moy)
- 2) **si** moy>=10 **alors** message ← "ADMIS"
 - Sinon**
 - Si** moy>=9 **alors** message ← "CONTROLE"
 - Sinon** message ← "REDOUBLE"
 - Finsi
 - FinSi
- 3) Ecrire(message)
- 4) Fin décision

Programme Pascal

```

program decision ;
uses wincrt ;
var moy : real ;
    message : string ;
Begin
write('Donner une moyenne') ; readln(Moy) ;
if moy>=10 then message:= 'ADMIS'
    else
        if moy>=9 then message := 'CONTROLE'
            else message := 'REDOUBLE' ;
Writeln(Message) ;
end.
```

Ex5 page 83 :

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

Corrigé :

Algorithme :

- 0) Début mots
- 1) Ecrire("Saisir une chaîne de caractère :"), lire(ch)
- 2) **si** pos(" ",ch)=0 **alors** écrire (" un seul mot")
 - sinon**
 - écrire (" Plusieurs mots")
 - Finsi
- 3) Fin Mots

T.D.O

Objet	Type/Nature	Rôle
Ch	chaîne	Chaîne à saisir

Exercice :

Écrire un programme qui permet de saisir un temps (heure et minute) lui ajoute 5minutes puis l'affiche.

Exemples :

heure : 10	heure : 10	heure : 23
minute : 20	minute:57	minute:55
après 5 minutes : 10:25	après 5 minutes : 11: 02	après 5 minutes : 00:00

Algorithme :

- 0) **Début** plus5min
- 1) Ecrire("Heure: "),lire(hh)
- 2) Ecrire("Minute: "),lire(mm)
- 3) **Si** mm<55 **alors** mm ← mm+5
 - | **sinon**
 - | **si** hh<23 **alors**
 - | | hh ← hh+1
 - | | mm ← mm-55
 - | | **sinon**
 - | | hh ← 0
 - | | mm ← mm-55
 - | **FinSi**
- FinSi**
- 4) Ecrire("Après 5 min : ", hh, " :", mm)
- 5) **Fin** Plus5min

Programme pascal :

```

program plus5min ;
uses winCRT ;
var hh,mm : integer ;
Begin
  write('Heure : ') ;readln(hh) ;
  write('minute : ') ;readln(mm) ;
  if mm<55 then mm := mm+5
    else
      if hh<>23 then
        begin
          hh:= hh+1 ;
          mm:= mm+5-60 ;
        end
      else
        begin
          hh:=0 ;
          mm := mm-55 ;
        end ;
  writeln('Après 5 min : ', hh, ' :', mm) ;
End.

```

Remarque :

Pour afficher l'heure et minute correctement (ajouter un zéro à gauche des chiffres <10 , exemple 05:08)

```
Var h,m :string[2] ;  
  
...  
Str(mm,m) ;  
Str(hh,h) ;  
If mm<10 then m := '0'+m ;  
If hh<10 then h :='0'+h ;  
writeln('Après 5 min : ', h, ' :', m) ;
```


T.D.O

Objet	Type/Nature	Rôle
Mois M	Entier Chaîne de caractère	Saisir le mois Contient le message à afficher

Pascal :

```

Program saison ;
uses wincrt ;
var mois:integer ;
    m : string ;
begin
  writeln('Donner le n° de mois') ; readln(mois) ;
  case mois of
    1,2,3 : M := 'saison hiver' ;
    4,5,6 : M:= 'saison printemps' ;
    7,8,9 : M:= 'saison été' ;
    10,11 : M:= 'saison automne' ;
    else M:= 'Erreur' ;
  End ;
  writeln(M) ;
End.

```

Exercice 1 page 102 :

Écrire un programme intitulé TOUCHE, qui affiche selon le cas, la nature du caractère (consonne, voyelle, chiffre ou symbole) correspondant à une touche saisie.

On considère que le clavier est verrouillé en minuscule.

Corrigé :

```

Program touche ;
uses wincrt ;
var
  nature : string ;
  c : char ;
begin
  writeln('Taper une touche') ;
  readln(c) ;
  case c of
    'a'..'z', 'A'..'Z': if upcase(c) in ['A', 'E', 'I', 'O', 'U', 'Y'] then
                        nature:='voyelle'
                      else nature:='consonne' ;
    '0'..'9' : nature:= 'Chiffre' ;
  else nature:='symbole' ;
  end ;
  writeln(nature) ;
end.

```

Exercice7 page 103 :

Écrire un programme qui permet d'afficher le nombre de jour d'un mois donné.

Corrigé :**Algorithme :**

- 0) Début mois
- 1) Ecrire("Donner le n° de mois"), lire(n)
- 2) **Selon n faire**
 - 1,3,5,7,8,10,12 : nb ← 31
 - 4,6,9,11 : nb ← 30
- 2 : Ecrire("Donner l'année "), lire(a)

Si (a mod 400 =0) ou ((a mod 4 =0) et (a mod 100 <>0)) alors nb ← 29

sinon nb ← 28

Finsi

FinSelon

- 3) Ecrire("Le nombre de jour est ",nb)
- 4) Fin mois

pascal :

```

Program mois ;
uses wincrt ;
var
n,a,nb : integer ;
begin

Write('Donner le n° de mois ') ; readln(n) ;
case n of
  1,3,5,7,8,10,12 : nb:= 31
  4,6,9,11 : nb:= 30
  2 : begin
    write('Donner l''année ');readln(a) ;
    if (a mod 400 =0) OR ( (a mod 4 =0) AND (a mod 100 <>0) ) then
      nb:= 29
    else nb:= 28 ;
  End ;
End ;
writeln('Le nombre de jour est ',nb)
End.

```

NB: Une année bissextile est une année comptant 366 jours au lieu de 365, c'est-à-dire une année comprenant un 29 février (exemple 2012)

sont bissextiles les années:

- ✓ divisibles par 4 mais non divisibles par 100
- ✓ divisibles par 400.

Applications : Voir activité 1 page 74-75 {or , bip chr(7) initialisation}

Activité 2

Traduire l'algorithme REGULATEUR en Pascal.

Traduction en Pascal

```
PROGRAM REGULATEUR;
USES WINCRT;
VAR
  Temp : INTEGER;
  Message : STRING;
  Bip : CHAR;

BEGIN
WRITE ('Entrer la valeur de la température : ');
READLN(Temp);
Bip := '';
Message := 'Régulateur non déclenché';
IF(Temp < 18) OR (Temp > 24)
  THEN
    BEGIN
      Message := 'Régulateur déclenché';
      Bip := Chr(7);
    END;
WRITE(Bip,Message);
END.
```

Exercice 6

Lire l'année et afficher si elle est éventuellement bissextile.

NB. Si l'année n'est pas divisible par 4, l'année n'est pas bissextile.

Exercice 9

Un chef d'entreprise décide d'automatiser les horaires de début de travail, de pause et de fin de travail. Le système consiste à déclencher une alarme et à afficher un message à des heures bien précises de la journée.

On se propose de faire un programme intitulé MINUTERIE qui lit l'heure maintenue par le système d'exploitation et affiche le message "Reprise du travail" et produit deux bips sonores si l'horaire est équivalent à 7 Heures et 30 Minutes ou 14 Heures, sinon affiche le message "Arrêt du travail" et produit un seul bip sonore si l'horaire est équivalent à 12 Heures ou 17 Heures et 15 Minutes.

NB.

- On rappelle que l'écriture de CHR(7) produit un bip sonore.
- La procédure prédéfinie **GETTIME** renvoie l'heure maintenue par le système d'exploitation.



Syntaxe: procedure **GETTIME**(var Heure, Minute, Seconde, Sec100: Word);
Description: les intervalles des valeurs renvoyées sont 0..23 pour heure, 0..59 pour minute, 0..59 pour Seconde et Sec100(centièmes de secondes).