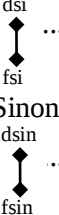
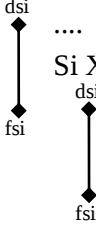


SOLUTION TD N°3 : LES STRUCTURES CONDITIONNELLES

Exercice 00 :

Répondez par « vrai » ou « faux » aux instructions suivantes et corriger les erreurs:

1) Si $N \leftarrow S$; Alors Faux  ... //affectation au lieu de condition	2) Si $A > B$ Faux  ... //il manque Alors	3) Si $N1 = 0$, $N2 = 0$ Alors Faux  ... //la virgule n'est pas un opérateur
4) Si $i \leftarrow i + 1$; Alors Faux  ... //affectation au lieu de condition	5) Si $A \leftarrow 10$ et $B \leftarrow 20$ Alors Faux  ... //affectation au lieu de condition	6) Si mois est 'mars' Alors Faux  ... //"est" n'est pas un opérateur
7) Si nbr est Positif Alors Faux  ... //nbr > 0	8) Si $A \lt> B$ et C Alors Faux  ... //A <> B et A <> C	9) Si $[N > 0]$ et $[N < 10]$ Alors Faux  ... //utiliser les parenthèses (...)
10) Si $X > Y$ Alors Faux  ... Sinon Alors //un seul Alors	11) Si $A > 0$ Alors Faux  ... Sinon  ... /* le sinon vient en dehors du bloc de Si ... Alors */	12) Si $X \geq 0$ Alors Faux  Si $X = 0$ Alors  /* le bloc interne doit être fermé avant la fermeture du bloc externe ou global */

Exercice 01 : Écrire un algorithme qui demande un entier à l'utilisateur, teste si ce nombre est positif (≥ 0) ou non, et affiche « positif » ou « négatif ».

- Modifier l'algorithme pour qu'il affiche « strictement positif », « nul » ou « strictement négatif »
- Cet exercice est corrigé durant le cours

Solution 1:

Algorithme :

```

Algorithme pos_neg;
Var A : entier ;

Début
    Ecrire('Donner un nombre A');
    Lire(N);
    Si (A >= 0) Alors
        Ecrire (A, ' est positif');
    Sinon
        Ecrire (A, ' est négatif');
    Fsi
Fin.
    
```

Solution 2:

Algorithme :

```
Algorithme pos_neg_nul;  
Var A : entier ;  
  
Début  
    Ecrire('Donner un nombre A');  
    Lire(N);  
    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.
```

Exercice 02 : Ecrire un algorithme qui calcule et affiche la valeur absolue d'un entier quelconque lu au clavier.

Solution :

Analyse :

Comprendre l'énoncé :

- " la valeur absolue " → (القيمة المطلقة)
- " entier quelconque lu au clavier " → déclarer une variable et la lire dans l'algorithme
 - identificateur : on va l'appeler, x
 - type : c'est mentionné dans l'énoncé (... entier ...) donc, c'est de type entier

Qu'est ce qui est demandé :

- Afficher –directement- la valeur absolue sur l'écran (utiliser "ecrire")

Trouver la solution :

- On teste sur la valeur du x :
 - ❖ Si c'est positif, on affiche le nombre x tel quel
 - ❖ Si c'est négatif on le multiplie par (-1) et on l'affiche

Algorithme :

```
Algorithme absolue;  
Var x : entier ;  
  
Début  
    Ecrire('Donner un nombre x');  
    Lire(x);  
    Si (x >= 0) Alors  
        Ecrire(' La valeur absolue de ', x, ' = ', x);  
    Sinon  
        Ecrire(' La valeur absolue de ', x, ' = ', (-1)*x);  
    Fsi  
Fin.
```

Exercice 03 : Ecrire l'algorithme qui permet de saisir un nombre puis déterminer s'il appartient à un intervalle donné, sachant que les extrémités de l'intervalle sont fixées par l'utilisateur.

Solution :

Analyse :

Comprendre l'énoncé :

- " saisir un nombre " → déclarer une variable et la lire dans l'algorithme
 - identificateur : on va l'appeler, x
 - type : ce n'est pas indiqué dans l'énoncé donc, on le choisit comme réel
- " un intervalle " → (المجال)
- " les extrémités " → (حدود المجال)
- " fixées par l'utilisateur " → l'utilisateur va donner les valeurs des extrémités donc, on va les lire dans deux variables :
 - identificateur : on va les appeler, a et b
 - type : ça sera des réels

Qu'est ce qui est demandé :

- "déterminer s'il appartient à l'intervalle" → vérifier s'il appartient ou non et l'afficher (rien à calculer)

Trouver la solution :

- On suppose que l'intervalle s'écrit comme ça : $[a, b]$
- On teste sur la valeur du x :
 - ❖ Si le x est supérieur ou égale à a et le x est inférieur ou égale à b donc il appartient.

Algorithme :

```
Algorithme  intervalle;
Var  a, b, x : réel ;

Début
    Ecrire('Donner les extrémités a et b :');
    Lire(a, b);
    Ecrire('Donner le nombre x:');
    Lire(x);

    Si  (x >= a et x <= b)  Alors
        Ecrire(x, ' appartient à l''intervalle');
    Sinon
        Ecrire(x, ' n''appartient pas à l''intervalle');
    Fsi
Fin.
```

Exercice 04 : Ecrire l'algorithme qui cherche et affiche le maximum et le minimum de trois valeurs entières saisies.

Solution 1:

Analyse :

Trouver la solution :

- Déclarer une variable qui va sauvegarder la valeur du minimum
- On suppose que le premier entier – A- est le minimum
- On compare ce minimum avec le reste des valeurs (B et C)
 - ❖ Si on trouve une valeur qui est inférieur au minimum, elle devient le nouveau minimum.
 - ❖ Sinon, on garde le minimum tel quel (on fait rien)
- Le même principe s'applique pour le maximum

Algorithme :

```

Algorithme minimum_maximum1;
Var A, B, C : entier ;
      min, max : entier ;
Début
    Ecrire('Donner les valeurs de A, B et C :');
    Lire(A, B, C);
    min ← A ; min ← A ;

    Si (B < min) Alors min ← B Fsi;
    Si (C < min) Alors min ← C Fsi;

    Si (B > max) Alors max ← B Fsi;
    Si (C > max) Alors max ← C Fsi;

    Ecrire('Le minimum = ', min);
    Ecrire('Le maximum = ', max);
Fin.

```

Solution 2:

Analyse :

Trouver la solution :

- Une variable est minimum de trois valeurs si elle est inférieure aux deux autres.
 - ❖ Donc, trouvez tous les cas possible pour que A soit minimum, B soit minimum et C soit minimum.

Algorithme :

```

Algorithme minimum_maximum2;
Var A, B, C : entier ;
Début
    Ecrire('Donner les valeurs de A, B et C :');
    Lire(A, B, C);

    // Calculer le min
    Si (A <= B et A <= C) Alors
      min ← A;
    Sinon
      Si (B <= A et B <= C) Alors
        min ← B;
      Sinon
        Si (C <= A et C <= B) Alors
          min ← C;
        Fsi
      Fsi
    Fsi

    // Calculer le max
    Si (A >= B et A >= C) Alors
      max ← A;
    Sinon
      Si (B >= A et B >= C) Alors
        max ← B;
      Sinon
        Si (C >= A et C >= B) Alors
          max ← C;
        Fsi
      Fsi
    Fsi

    Ecrire('Le minimum = ', min);
    Ecrire('Le maximum = ', max);
Fin.

```

Exercice 05 : (Examen 2013/2014) : Etant donné une somme d'argent donnée. Ecrire un algorithme qui permet d'obtenir la suite des pièces totalisant la somme donnée. Les espèces disponibles sont des pièces de 50, 20, 10, 5 et 1 Dinars. (L'algorithme doit donner le nombre minimal de pièces).

Solution :

Analyse :

➤

Algorithme :

```

Algorithme monnaie ;
Var Somme, P1, P2, P3, P4, P5: entier ;
Début
    Lire (Somme);
    Si (Somme >= 50) Alors
        P1 := somme DIV 50;
        Somme := somme MOD 50;
    Sinon
        Si (Somme >= 20) Alors
            P2 := somme DIV 20;
            Somme := somme MOD 20;
        Sinon
            Si (Somme >= 10) Alors
                P3 := somme DIV 10;
                Somme := somme MOD 10;
            Sinon
                Si (Somme >= 5) Alors
                    P4 := somme DIV 5;
                    Somme := somme MOD 5;
                Sinon
                    P5 := somme;
                Finsi
            Finsi
        Finsi
    Finsi
    Ecrire ('La suite des pièces est :',P1,' pièces de 50 DA et ',P2,
    ' pièces de 20 DA et ',P3,' pièces de 10 DA et ',P4,' pièces de 5 DA et '
    ,P5,' pièces de 1 DA') ;
Fin.

```

Exercice 06 : Ecrire l'algorithme qui permet de saisir le jour, le mois et l'année d'une date (le mois et donné en numéro), de déterminer si c'est une date correcte et de l'afficher en transformant le numéro du mois en nom du mois (par exemple 03 mars 2012)

Remarque : utiliser la structure selon pour la dernière partie.

Exercice 07 : (Examen 2013/2014) : Écrire un algorithme qui demande une date sous la forme de 3 nombres entiers (jour, numéro du mois et l'année) et affiche la saison (ex : 12/02/2008 ; hiver de 2008). (On supposera que le premier jour de la saison est toujours le 21)

Solution :

Analyse :

➤

➤ On suppose que la date est valide (les jours sont compris entre 1 et 31)

Algorithme :

```
Algorithme saisons;
Var jour, mois, annee : entier ;
Début
  Lire (jour, mois, annee) ;

  Si (mois=12 et jour >= 21) ou mois=1 ou mois=2 ou (mois=3 et jour<21) Alors
    Ecrire ('Hiver de ', annee) ;
  Sinon
    Si (mois=3 et jour >= 21) ou mois=4 ou mois=5 ou (mois=6 et jour<21) Alors
      Ecrire ('Printemps de ', annee) ;
    Sinon
      Si (mois=6 et jour >= 21) ou mois=7 ou mois=8 ou (mois=9 et jour<21) Alors
        Ecrire ('été de ', annee) ;
      Sinon
        Ecrire ('Automne de ', annee) ;
      Finsi
    Finsi
  Finsi
Fin.
```

Exercice 08 : Une année bissextile (contient 366 jours) si elle est multiple de 4, sauf les années de début de siècle (qui se terminent par 00) qui ne sont pas bissextile que si elles sont divisibles par 400.

- Exemple :**
- 1980 et 1996 sont bissextiles car elles sont divisibles par 4.
 - 2000 est une année bissextile car elle est divisible par 400.
 - 2100 et 3000 ne sont pas bissextiles car elles ne sont pas divisibles par 400.

Ecrire un algorithme qui permet de déterminer si un entier positif donné correspond à une année bissextile ou non.

Solution 1:

Analyse :

➤

Algorithme :

```
Algorithme bissextiles;
Var n : entier ;
Début
  Lire (n) ;

  Si (n Mod 4 = 0) Alors
    Si (n Mod 100 = 0) Alors
      Si (n Mod 400 = 0) Alors
        Ecrire (n, ' est bissextile') ;
      Sinon
        Ecrire (n, ' est non bissextile') ;
      Finsi;
    Sinon
      Ecrire (n, ' est bissextile') ;
    Finsi
  Sinon
    Ecrire (n, ' est non bissextile') ;
  Finsi
Fin.
```

Solution 2:

Analyse :

➤

Algorithme :

```
Algorithme bissextiles;  
Var n : entier ;  
Début  
  Lire (n) ;  
  
  Si (n Mod 400 = 0) OU ((n Mod 100 <> 0) ET (n Mod 4 = 0)) Alors  
    Ecrire (n, ' est bissextile') ;  
  Sinon  
    Ecrire (n, ' est non bissextile') ;  
  Finsi  
Fin.
```

Solution 3:

Analyse :

➤

Algorithme :

```
Algorithme bissextiles;  
Var n : entier ;  
Début  
  Lire (n) ;  
  
  Si (n Mod 4 = 0) ET ((n Mod 100 <> 0) OU (n Mod 400 = 0)) Alors  
    Ecrire (n, ' est bissextile') ;  
  Sinon  
    Ecrire (n, ' est non bissextile') ;  
  Finsi  
Fin.
```

Exercice 09 : Ecrire un algorithme qui demande trois noms à l'utilisateur et l'informe ensuite s'ils sont rangés ou non dans l'ordre alphabétique.

Solution :

Analyse :

➤

Algorithme :

```
Algorithme noms;  
Var a, b, c : chaines ;  
Début  
  Ecrire("Entrez successivement trois noms : ") ;  
  Lire (a, b, c) ;  
  
  Si (a < b) ET (b < c) Alors  
    Ecrire (' Ces noms sont classés alphabétiquement') ;  
  Sinon  
    Ecrire (' Ces noms ne sont pas classés') ;  
  Finsi  
Fin.
```

Exercice 10 : Soient 3 chaines de caractères c1, c2, c3. Déterminer et afficher la chaine de caractères c4 par concaténation de la plus petite et la plus grande chaine parmi c1, c2, c3.