

TD N°2 : Les structures conditionnelles et alternatives

(Correction)

Exercice 1

Ecrire un algorithme qui demande un nombre entier à l'utilisateur et affiche si le nombre est pair ou impair.

Algorithme Pair_impair;

Var

nombre: Entier;

Début

Ecrire ("Donnez-moi un nombre");

Lire (nombre);

Si (nombreMOD2=0) alors

Ecrire ("c'est un nombre pair");

Sinon

Ecrire ("c'est un nombre impair");

FinSi

Fin

Exercice 2

Ecrire un algorithme qui permet de lire trois valeurs entières a, b et c et d'afficher le maximum des trois nombres.

Version avec la structure conditionnelle simple (un seul choix)

Algorithme Maximum;

var

a, b, c, max : entier

Début

Ecrire ("Donner la valeur de a");

Lire (a);

Ecrire ("Donner la valeur de b");

Lire (b);

Ecrire ("Donner la valeur de c");

Lire (c);

max ← a

Si (max < b) alors

max ← b;

FinSi

Si (max < c) alors

max ← c;

FinSi

Ecrire("Le max est ", max);

Fin

Version avec la structure conditionnelle imbriquée

Algorithme Maximum2;

var

a, b, c, Max: entier;

Début

Ecrire ("Donner la valeur de a");

Lire (a);

Ecrire ("Donner la valeur de b");

Lire (b);

Ecrire ("Donner la valeur de c");

Lire (c);

Si (b <= a ET c<=a) alors

Max ← a;

Sinon

Si (a <=b ET c<=b) alors

Max ← b;

Sinon

Si (a <=c ET b<=cc) alors

Max ← c;

FinSi

Fin Si

Fin Si

Ecrire("Le max est " + max);

Fin

Version avec la structure conditionnelle imbriquée

Algorithme Maximum3;

var

a, b, c, Max: entier;

Début

Ecrire ("Donner la valeur de a");

Lire (a);

Ecrire ("Donner la valeur de b");

Lire (b);

Ecrire ("Donner la valeur de c");

Lire (c);

Si (b <a et c<a) alors

Max ← a;

Sinon

Si (c<b) alors

Max ← b;

Sinon

Max ← c;

FinSi

FinSi

Ecrire("Le max est " + max)

Fin

Exercice 3

Ecrire un algorithme qui permet la résolution dans \mathbb{R} d'une équation du premier degré : $ax + b = 0$, avec a et b sont des réels.

Algorithme EquationpremierDegre;

```
Var a,b,c x: réels;
Début
    Ecrire( "saisir les valeurs des nombres a et b");
    Lire (a,b);
    Si (a=0) alors
        Si (b=0) alors
            Ecrire ("Résultat: Infinité de solutions");
        Sinon
            Ecrire ("Résultat: Aucune solution");
        Sinon
            x< -b/b;
            Ecrire ("Résultat: x=", x);
            Ecrire ("Vérification:", a, "*",x,"+",b,"=", a*x+b);
    Finsi
Fin
```

Exercice 4

Les étudiants ayant passé l'examen d'algorithmique en session de Juin ont été classés selon leurs notes en trois catégories :

- Pour une note inférieure ou égale à 5, l'étudiant est éliminé,
- Pour une note supérieure ou égale à 5 et inférieur strictement à 10, l'étudiant passe la session de rattrapage,
- Pour une note supérieure ou égale à 10, l'étudiant valide le module.

Ecrire un algorithme qui demande à l'utilisateur d'entrer la note du module, puis affiche la situation de l'étudiant (on suppose que l'utilisateur saisie une note valide entre 0 et 20).

1^{ère} version :

Algorithme NotesMod ;

```
Var note : réel ;
Début
    Ecrire("Entrez la note du module :)");
    Lire (note) ;
    Si (note < 5) alors
        Ecrire ("l'étudiant est éliminé") ;
    Sinon Si (note <10) alors
        Ecrire ("l'étudiant passe en rattrapage") ;
    Sinon
        Ecrire ("l'étudiant a validé le module");
    Finsi
Finsi
```

Fin

2^{ème} version:

Algorithme NotesMod ;

Var note : réel ;

Début

Ecrire("Entrez la note du module") ;

Lire (note) ;

Si (note < 5) **alors**

Ecrire("l'étudiant est éliminé") ;

Finsi

Si (note >= 5 **et** note < 10) **alors**

Ecrire(" l'étudiant passe en rattrapage") ;

Finsi

Si (note > 10) **alors**

Ecrire("l'étudiant a validé le module") ;

Finsi

Fin

Exercice 5

Ecrire un algorithme qui demande l'âge d'un enfant à l'utilisateur. Ensuite, il l'informe de sa catégorie:

- "Poussin" de 6 à 7 ans
- "Pupille" de 8 à 9 ans
- "Minime" de 10 à 11 ans
- "Cadet" après 12 ans

Algorithme Catégorie_Enfant;

Var

age: entier;

Debut

Ecrire ("Entrez l'âge de l'enfant : ");

Lire (age);

Si (age >= 12) **Alors**

Ecrire ("Catégorie Cadet");

Sinon Si (age >= 10) **Alors**

Ecrire ("Catégorie Minime");

Sinon Si age >= 8 **Alors**

Ecrire ("Catégorie Pupille");

Sinon Si (age >= 6) **Alors**

Ecrire ("Catégorie Poussin");

Finsi

Finsi

Finsi

Finsi

Fin

Exercice 6:

Un magasin de reprographie facture 1 DH les dix premières photocopies, 0,6 DH les vingt suivantes et 0,4 DH au-delà.

Ecrivez un algorithme qui demande à l'utilisateur le nombre de photocopies effectuées et qui affiche la facture correspondante.

Algorithme Photocopies;

Var

n : entier;

prix : réel;

Début

ecrire("Nombre de photocopies : ");

lire(n);

si ($n \leq 10$) alors

prix $\leftarrow n \times 1$;

sinon si ($n \leq 30$) alors

prix $\leftarrow 10 \times 1 + (n - 10) \times 0.6$;

sinon

prix $\leftarrow 10 \times 1 + 20 \times 0.6 + (n - 30) \times 0.4$;

finsi

finsi

ecrire("Le prix total est: ", prix);

Fin

Exercice 7

Les habitants de la ville XX paient l'impôt selon les règles suivantes :

- les hommes de plus de 20 ans paient l'impôt
- les femmes paient l'impôt si elles ont entre 18 et 35 ans
- les autres ne paient pas d'impôt

L'algorithme demandera l'âge et le sexe, et se prononcera donc ensuite sur le fait que l'habitant est imposable ou non.

Nous utilisons dans cette correction le type caractère afin de stocker le sexe de l'habitant: 'M' pour masculin et 'F' pour féminin. Ne pas oublier les côtes ' lors de l'utilisation des caractères.

Algorithme Population;

Var

sexe: caractère;

age: entiere;

c1, c2: booléen;

Début

ecrire("Entrez le sexe (M/F) : ")

lire(sexe);

ecrire("Entrez l'âge: ");

lire(age);

c1 \leftarrow (sexe = 'M' ET age > 20);

```

c2 ← (sexe = 'F' ET (age > 18 ET age < 35));
Si (c1 OU c2) alors
    écrire("Imposable");
Sinon
    écrire("Non Imposable");
Finsi

```

Fin

Exercice 8

A la caisse d'un supermarché, nous bénéficions d'une remise de 1% sur le montant d'un achat compris entre 300 et 750 euros et 2% au-delà de 750 euros.

Ecrire un algorithme qui après lecture du montant initialement du, affiche le montant à payer.

Algorithme Remise

```

Var
    montant, TauxRemise: réel;
Début
    Ecrire ("Donner le montant initial d'achat");
    lire(montant);
    Si (montant > 750) alors
        tauxRemise ← 0.02 ;
    Sinon
        Si (montant>300) alors
            tauxRemise ← 0.01 ;
        Sinon
            tauxRemise ← 0 ;
    FinSi
    Finsi
    montant ← montant * (1-tauxRemise) ;
    Ecrire ("Vous devez payer le montant : ", montant)

```

Fin

Exercice 9:

Ecrire un algorithme qui affiche la ou les solution d'une équation du second degré de la forme $ax^2+bx+c=0$.

Algorithme EquationSecondDgr;

```

Var a,b,c,x,x1,x2,delta: réels;
Début
    Ecrire( "saisir les valeurs des nombres a,b et c");
    Lire (a,b,c);
    Si (a=0) ET (b=0) ET (c=0) alors
        Ecrire( "Tous réel est solution de l'équation");
    Sinon Si (a=0) ET (b=0) alors
        Ecrire( " Impossible ");
    Sinon Si (a=0) alors
        Ecrire (" résolution de l'équation de premier degré bx+c=0");
        x ← -c/b;
        Ecrire (" x=", x);
    Sinon
        delta ← b*b-4*a*c;

```

```

    Si (delta<0) alors
      Ecrire (" pas de solution dans R");
    Sinon si (delta=0) alors
      Ecrire ("l'équation à une solution réel unique dans R");
       $x \leftarrow -b/(2*a)$ ;
      Ecrire (" x=", x);
    Sinon
       $x1 \leftarrow (-b+racine(delta))/(2*a)$ ;
       $x2 \leftarrow (-b-racine(delta))/(2*a)$ ;
      Ecrire ("les deux solutions dans R sont x1=", x1, "et x2=",x2);
    Finsi
  Finsi
Finsi
Fin

```

Exercice 10

Ecrire un algorithme qui permet de déterminer la nature d'un triangle à partir de ses cotés. Les triangles se distinguent selon la longueur des côtés :

- Trois côtés de même longueur : **triangle équilatéral**
- Deux cotés égaux : **triangle isocèle**
- Trois côtés de longueur différentes : **triangle quelconque**
- Triangle avec angle droit : **triangle rectangle**
- Triangle avec angle droit plus deux cotés égaux : **triangle rectangle isocèle**

La condition:

$(a+b > c)$ et $(b+c > a)$ et $(a+c > b)$ est une condition nécessaire pour que le triplet de réel (a, b, c) corresponde aux trois côtés d'un triangle.

Algorithme natureTriangle ;

Var a,b,c : réel ;

A_carre, B_carre, C_carre : reel ;

Début

Ecrire("Détermination de la nature du triangle en fonction de a,b,c") ;

Ecrire("Entrer les côtés a,b et c") ;

Lire (a,b,c) ;

Si ($a < 0$) ou ($b < 0$) ou ($c < 0$) ou ($a+b < c$) ou ($b+c < a$) ou ($a+c < b$) **Alors**

Ecrire("Ces longueurs ne peuvent constituer les côtés d'un triangle") ;

Sinon

Si(($a=b$) et ($b=c$) et ($c=a$)) **Alors**

Ecrire("Triangle équilatéral");

Sinon

Si ($a+c=b$) ou ($a+b=c$) ou ($b+c=a$) **Alors**

Ecrire("Triangle Plat");

Sinon

```

A_carre ← a*a;
B_carre ← b*b;
C_carre ← c*c;
  Si ((A_carre=B_carre+C_carre) ou
      (B_carre=A_carre+C_carre) ou
      (C_carre=B_carre+A_carre)) Alors
    Si ((a=b) ou (b=c) ou (c=a))Alors
      Ecrire("Triangle rectangle isocèle");
    Sinon
      Ecrire("Triangle rectangle") ;finsi
  Sinon
    Si((a=b)ou (a=c)ou (b=c))Alors
      Ecrire("Triangle isocèle");
    Sinon
      Ecrire("Triangle quelconque");
  Finsi
Finsi

```

Fin

Exercice 11

Écrire un algorithme qui lit un caractère au clavier puis affiche s'il s'agit d'une lettre minuscule, d'une lettre majuscule, d'un chiffre ou d'un caractère spécial.

Algorithme nature_car ;

Var

c : Caractère ;

Début

Ecrire("Entrer un caractère : ") ;

Lire(c) ;

Selon c Faire

"a".."z" : Ecrire("le caractère",c, "est une lettre minuscule") ;

"A".."Z" : Ecrire("le caractère",c, "est une lettre majuscule") ;

"0".."9" : Ecrire("le caractère",c, "est un chiffre") ;

Sinon

Ecrire("le caractère",c, "est un caractère spécial") ;

FinSelon

Fin

Exercice 12

Écrire un algorithme qui lit un chiffre et qui affiche son nom en lettres et le mois correspondant.

Algorithme Chiffre_nom_mois;

Var

n: entier;

Début

Ecrire ("Donnez un chiffre entre 1 et 12");

lire (n);

Selon que n vaut

- 1: Ecrire ("Mois un: Janvier");
 - 2: Ecrire ("Mois deux: Février");
 - 3: Ecrire ("Mois trois: Mars");
 - 4: Ecrire ("Mois quatre: Avril");
 - 5: Ecrire ("Mois cinq: Mai");
 - 6: Ecrire ("Mois six: Juin");
 - 7: Ecrire ("Mois Sept: Juillet");
 - 8: Ecrire ("Mois huit: Aout");
 - 9: Ecrire ("Mois neuf: Septembre");
 - 10: Ecrire ("Mois dix: Octobre");
 - 11: Ecrire ("Mois Onze: Novembre");
 - 12: Ecrire ("Mois douze: Décembre");
- Autrement: Ecrire ("Vous n'avez pas choisi entre 1 et 12");

Fin selon

Fin