



Correction de la série TD 2

Exercice 1

On introduit une variable **etat** de type chaîne de caractères.

On va imbriquer les tests de la manière suivante :

```
si (t < 0) alors etat ← "glace " ;  
sinon  
    si (t <= 100) alors etat ← "liquide" ;  
    sinon etat ← "vapeur" ;  
finsi  
finsi
```

On commence par évaluer la première condition ($t < 0$). Si la condition est vérifiée, on effectue l'affectation $etat \leftarrow "glace"$; sinon ($t \geq 0$), on évalue la deuxième condition ($t \leq 100$) qui en fait est équivalente à ($t \geq 0$) and ($t \leq 100$). Si la condition est vérifiée, on effectue l'affectation $etat \leftarrow "liquide"$; sinon ($t > 100$), on effectue l'affectation $etat \leftarrow "vapeur"$.

Algorithme Etat_Eau ;

Variables t : Entier ;

etat : chaîne de caractères ;

Début

Ecrire ("Entrer une température=") ; Lire(t) ;

```
si (t < 0) alors etat ← "glace " ;
```

```
sinon
```

```
    si (t <= 100) alors etat ← "liquide" ;
```

```
    sinon etat ← "vapeur" ;
```

```
finsi
```

```
finsi
```

Ecrire ("Etat de l'eau correspondant à la température ", t, " est :", etat) ;

Fin.

Exercice 2

On introduit une variable mention de type chaîne de caractères.

On peut utiliser une alternative multiple pour déterminer la mention au bac en fonction de la note de l'étudiant :

```
si (note < 10) alors mention ← "insuffisant" ;
sinon si (note < 12) alors mention ← "passable" ;
    sinon si (note < 14) alors mention ← "assez bien" ;
        sinon si (note < 16) alors mention ← "bien" ;
            sinon mention ← "très bien" ;
            finsi
        finsi
    finsi
finsi
```

Algorithme Ment_Bac;

Variables note : réel ;
mention : chaîne de caractères ;

Début

Ecrire ("Entrer une note entre 0 et 20 :") ; Lire(note) ;

```
si (note < 10) alors mention ← "insuffisant" ;
sinon si (note < 12) alors mention ← "passable" ;
    sinon si (note < 14) alors mention ← "assez bien" ;
        sinon si (note < 16) alors mention ← "bien" ;
            sinon mention ← "très bien" ;
            finsi
        finsi
    finsi
finsi
```

finsi

Ecrire ("La mention du Bac correspondante à la note ", note, " est :", mention) ;

Fin.

Remarques : L'algorithme précédent a considéré les notes dans l'ordre croissant. Rédigez une autre version qui permet de considérer les notes dans l'ordre décroissant. Explorez d'autres possibilités.

Exercice 3

Algorithme Categ_Sport;

Variables age : entier ;
categorie : chaîne de caractères ;

Début

Ecrire ("Entrer un age entre 6 et 14 :") ;

Lire(age) ;

```

si ((age < 6) ou (age > 14)) alors écrire ("enfant avec un âge non traité ") ;
sinon si (age <= 7) alors  categorie ← "Poussin" ;
    sinon si (age <= 9) alors categorie ← "Pupille" ;
        sinon si (age <= 11) alors categorie ← "Minime" ;
            sinon  categorie ← "Cadet" ;
            finsi
        finsi
    finsi
fini

Ecrire ("La catégorie correspondante à l'âge ", age, " est :", categorie) ;

Fin.

```

Exercice 4

```

Algorithme Val_Abs
Variables x, va : Entier ;
Début
Ecrire ("Entrer un entier=") ; Lire(x) ;
Si ( x >= 0 ) Alors va ← x ;
    Sinon va ← -x ;
FinSi

Ecrire ("|", x, "|=", va) ;

Fin.

```

Exercice 5

```

Algorithme pair_impair ;
Variable n : Entier ;
Début

Ecrire("Entrer un entier=") ; Lire(n) ;
Si ( n mod 2 == 0 ) Alors Ecrire ("c'est un entier pair") ;
Sinon Ecrire ("c'est un entier impair") ;
FinSi

Fin.

```

Exercice 6

Algorithme equa2d ;

Variables a, b, c, delta, x1, x2 : Réel ;

Début

Ecrire ("Entrer la valeur de a (non nulle)=") ; Lire(a) ;

Ecrire ("Entrer la valeur de b=") ; Lire(b) ;

Ecrire ("Entrer la valeur de c=") ; Lire(c) ;

//traitement

delta \leftarrow b*b - 4*a*c ;

Si (delta \geq 0) Alors

 Si (delta == 0) Alors x1 \leftarrow -b/(2*a) ; x2 \leftarrow -b/(2*a) ;

 Sinon x1 \leftarrow (-b- racine(delta))/(2*a) ;

 x2 \leftarrow (-b+ racine(delta))/(2*a) ;

 Finsi

Finsi

//Affichage

Si(delta<0) alors Ecrire ("Pas de solution dans R ") ;

Sinon Ecrire (" La solution x1 = ", x1) ;

 Ecrire (" La solution x2 = ", x2) ;

Fin

Remarque : Dans cet algorithme, on suppose que l'utilisateur va toujours entrer une valeur de a non nulle. Sinon ce n'est pas une équation du second degré. On suppose aussi qu'il existe une fonction **racine(x)** qui permet de retourner la racine carrée d'une variable x.

Exercice 7

Algorithme calculatrice ;

Variables val1, val2 : Réel ;

 operation : caractère ;

Début

Ecrire ("Première opérande=") ; Lire (val1) ;

Ecrire ("Opération=") ; Lire (operation) ;

Ecrire ("Deuxième opérande=") ; Lire (val2) ;

Cas où operation vaut

 "+" : Ecrire ("Résultat =", val1 + val2) ;

 "-" : Ecrire ("Résultat =", val1 - val2) ;

 "*" : Ecrire ("Résultat =", val1 * val2) ;

 "/" : Si (val2 \neq 0) Alors Ecrire ("Résultat =", val1 / val2) ;

 Sinon Ecrire ("Division par zéro!") ;

 FinSi

Autre : Ecrire ("opérateur erroné...")

FinCas

Fin

Exercice 8

```
Algorithme Nature_Caractère ;  
Variables c: caractère ;  
Début  
Ecrire("Entrer un caractère=") ; Lire(c) ;  
Cas où c vaut  
'a'..'z' : Ecrire ("C est une lettre miniscule") ;  
'A'..'Z' : Ecrire ("C'est une lettre majuscule") ;  
'0'..'9' : Ecrire ("C'est un chiffre") ;  
Autre : Ecrire ("c'est un caractère spécial") ;  
finCas  
Fin
```

Exercice 9

```
Algorithme Bissextile ;  
Variables n : Entier ;  
Début  
  
Ecrire ("Entrer l'année=") ; Lire(n) ;  
Si ( n mod 400 == 0) ou (( n mod 100 != 0) et (n mod 4== 0)) Alors  
    Ecrire ("Année bissextile") ;  
Finsi  
  
Fin
```