

**Examen du baccalauréat 2014 : Session Principale**  
**Corrigé de l'épreuve d'Algorithmique et de Programmation - SI**

**Exercice 1 : ( 2,25 pts = 0.25 x 9 )**

Valider chacune des propositions suivantes en mettant dans la case correspondante la lettre **V** si elle est correcte ou la lettre **F** sinon.

1- Après exécution du programme **Jeux** ci-dessus pour  $N=1$ , le message affiché sera :

F "Balle sous gobelet 1"       V "Balle sous gobelet 2"       F "Balle sous gobelet 3"

2- Soit **f** un fichier de booléens.

a- Pour remplir un champ de la variable **g3**, on peut utiliser l'instruction :

F Lire (g3.balle)       V Lire (g3.couleur)       V Lire (f, g3.balle)

b- Pour afficher le contenu de la variable **g1**, on peut utiliser l'instruction :

F Ecrire (g1)       V Ecrire (g1.balle, g1.couleur)       F Ecrire (f, g1.balle)

**Exercice 2 : (2,75 points)**

0) DEF FN Rech\_cycle (**h, rang : entier**) : **entier**

1) Si (**h = 4**) Alors **Rech\_cycle** ← **rang**

    Sinon Si (**h mod 2 = 0**) Alors

**Rech\_cycle** ← FN Rech\_cycle(**h div 2, rang+1**)

    Sinon **Rech\_cycle** ← FN Rech\_cycle(**3\*h+1, rang+1**)

    FinSi

2) Fin FN Rech\_cycle

**Exercice 3 : (4 points)**

0) DEF FN calcul\_pgcd (**M : Mat ; N : entier**): **entier**

1) Pour i de 2 à N faire

    Pour j de 1 à N-i+1 faire

**M[i, j]** ← FN pgcd (**M[i-1, j], M[i-1, j+1]**)

    Fin Pour

Fin Pour

2) **calcul\_pgcd** ← **M[N, 1]**

3) Fin calcul\_pgcd

0) DEF FN pgcd (a, b: entier) : entier  
 1) Si (a = b) alors pgcd ← a  
     Sinon si (a > b) alors pgcd ← FN pgcd (a-b, b)  
     Sinon pgcd ← FN pgcd (a, b-a)  
 FinSi  
 2) Fin pgcd

### Problème : (11 points)

#### Analyse du programme principal :

Nom : Calcul\_M\_Molaire  
 Résultat = fr  
 fr = [Assigner(fr, "C:\Resultats.dat")] PROC Stocker (fa, fr, fm)  
 fa = [Assigner(fa, "C:\Atomes.dat") ] PROC Remplir (fa)  
 fm= [Assigner( fm, "C:\Molecules.txt")]  
 Fin Calcul\_M\_Molaire

#### T.D.N.T

Types
Atome_mol = Enregistrement
Nom : Chaîne
Masse : réel
Fin Atome_mol
Masse_atome_mol = Fichier d' Atome_mol

#### T.D.O.G.

Objet	Type/Nature
fa,fr	Masse_atome_mol
fm	Text
Stocker	Procédure
Remplir	Procédure

#### Analyse de la procédure Stocker :

DEF PROC Stocker (var fma, R: Masse\_atome\_mol; var f:texte)  
 Resultat =R  
 R= [Ouvrir (fma), Ouvrir (f), Recréer (R)]  
 Tant que non(fin\_fichier (f)) faire  
     Lire\_nl(f, ch)  
     mm.nom ← Sous chaine(ch,1,pos(";",ch)-1)  
     Efface (ch,1,pos(";",ch))  
     mm.masse ← FN Calcul( ch , fma )  
     Ecrire(R,mm)  
 FinTantque  
 Fermer(f)  
 Fermer(R)  
 Fermer(fma)  
 Fin Stocker

## T.D.O.L.

Objet	Type/Nature
Ch	Chaîne
mm	Atome_mol
Calcul	Fonction

### Analyse de la fonction Calcul :

```
Def Fn Calcul (ch : chaîne ; var fa : masse_atome_mol) : Réel
Resultat = Calcul ← M
M = [M←0] Répéter
  Atome ←ch[1]
  Efface(ch,1,1)
  Si(ch[1] dans ["a".."z"]) ET (ch<>"") Alors
    atome←atome + ch[1]
    Efface(ch,1,1)
  Fin Si
[nb ← 0] Tant que (ch[1] dans ["0".."9"]) et (ch<>"") Faire
  Valeur (ch[1], x, e)
  nb← nb*10 + x
  Efface (ch,1,1)
Fin tant que
Si (nb=0) Alors
  nb← 1
Fin Si
m ←m + nb * FN masse_at(fa,atome)
Jusqu'à (ch = "")
Fin Calcul
```

## T.D.O.L.

Objet	Type/Nature
m	Réel
nb ,x, e	Entier
atome	Chaîne[2]
masse_at	Fonction

### Analyse de la fonction masse\_at :

```
DEF FN masse_at (var f : Masse_atome_mol ; nomAt : Chaîne) : réel
Résultat = masse_at ← M
M = [Ouvrir(f) , Trouve ← faux ]
Tant que (NON (Fin_fichier(f) )) et ( NON (Trouve)) Faire
  Lire (f, At)
  Si (At.nom = nomAt) Alors
    Trouve ← vrai
    M ←At.masse
  FinSi
FinTantque
Fermer (f)
Fin masse_at
```

## TDOL

Objet	Type/Nature
M	Réel
Trouve	Booléen
At	Atome_mol

### Analyse de la procédure Remplir :

DEF PROC Remplir (var fma: Masse\_atome\_mol)

Resultat =fma

fma= [Recréer (fma)]

Pour i de 1 à N faire

    Avec atome faire

        Répéter

            nom = Donnée ("Nom atome : ")

            Jusqu'à ((Long(nom) =1) ET (nom[1] dans ["A".."Z"])) ou

            ((Long (nom)=2) ET (nom[1] DANS ["A".."Z"]) ET(nom[2] DANS ["a".."z"]))

            Répéter

                Masse = donnée ("Masse atome : ")

                Jusqu'à (Masse >0)

        Fin Avec

        Ecrire (fma,atome)

Fin pour

Fermer(fma)

N = Répéter

    N=donnée ("Nombre d'atomes : ")

Jusqu'à N dans [1..50]

Fin Remplir

### T.D.O.L.

Objet	Type/Nature
N	Octet
atome	Atome_mol