

1) Exercice sur le cours (8)

Soit le programme VB :

```
Dim x As Integer
Dim y As Integer
Private Sub calcul (ByRef R, ByVal V As Integer)
    R = V + 2
    V = R + V
    MsgBox "Dans Sous-programme, R = " & R & " V = " & V
End sub
End Sub

Sub Private Sub Form_Load()
    x = 1: y = 4
    Call calcul (x,y) :
    MsgBox "Dans Prog-Principal, X = " & x & " Y = " & y
    x = 0 : y = 6
    Call calcul (x, y) :
    MsgBox "Dans Prog-Principal, X = " & x & " Y = " & y
    x = 4,
    Call calcul (x,-3) :
    MsgBox "Dans Prog-Principal, X = " & x & " Y = " & y
End Sub
```

- a) Ce programme fait 6 écritures (message box) Lesquelles ? justifiez.
- b) Détailler le fonctionnement du premier appel à Calcul (x,y).

2) Problème :

On veut manipuler des « fourmis » qui ont chacune une position (x, y), une vitesse (Vx, Vy), une classe et une quantité de nourriture transportée (sa charge). La classe de la fourmi peut être nourrice, soldate, ouvrière ou reine.

- a) Déclarer les types « position », « vitesse » et « fourmi » (Fourmi dans la suite) et le type énuméré « classe ». (2,5 points)
- b) Déclarer un type « Ensemble de fourmis » (EnsF dans la suite) contenant un tableau de « fourmi » de taille NMAX et un entier représentant le nombre de fourmis dans cet ensemble de fourmis. NMAX est une variable globale (0,5 points)

Remarque : Pour la suite de l'exercice, on considère que tous les éléments du tableau de fourmis ont été déjà initialisés.

Ecrire en VB les sous-programmes suivants :

- c) Une fonction qui renvoie la distance entre deux fourmis ; (2 point)
- d) Un sous-programme qui recherche dans un ensemble de fourmis la fourmi qui est la plus chargée ; (2 points)
- e) Un sous-programme qui prend une fourmi et un ensemble de fourmis ordonné par charge croissante et insère la fourmi dans l'ensemble de fourmis de manière à conserver un ensemble de fourmis ordonné. Le tableau résultat doit donc rester ordonné. (5 points)

```
Sub insertFourmi (ByVal fourmi As Fourmi, ByRef ensFourmi As EnsF)
```