

Exercice I :

1 - Trouver le résultat fourni par l'algorithme :

```
Procédure SomCar ( →X1 : numérique, → X2 : numérique, ↔S : numérique)
```

```
Début
```

```
    X1 ← X1 * X1
```

```
    X2 ← X2 * X2
```

```
    S ← X1 + X2
```

```
Fin
```

```
Programme principal :
```

```
    Variables X, Y, Z : numériques
```

```
    X ← 3
```

```
    Y ← 4
```

```
    Z ← 0
```

```
    SomCar(X, Y, Z)
```

```
    Ecrire X, « ^2 + », Y, « ^2 = », Z
```

```
Fin du programme principal
```

2 - Remplacer dans ce programme la procédure par une fonction.

Exercice II : Une procédure est déclarée par :

```
Procédure Essai (→A : numérique, ↔B : numérique, ↔ C : numérique)
```

```
Début
```

```
    A ← A + 1
```

```
    B ← 22
```

```
    C ← C + 3
```

```
Fin
```

Parmi les appels suivants certains ne sont pas corrects, expliquer pourquoi. Pour les autres, trouver les valeurs des paramètres A, B, C au début et à la fin de son exécution, des variables X, Y et Z ensuite.

Avant chacun des appels, on effectue :

```
X ← 3
```

```
Y ← 7
```

```
Z ← 11
```

1 – Essai (1, 2, 3)

2 – Essai (X, Y, Z)

3 – Essai (Z, Y, X)

4 – Essai (1+X*10, Y, Z)

5 – Essai (X, X, Z)

6 – Essai (X, Y, Y)

Exercice III : Quels sont les résultats produits par l'algorithme suivant :

```
Procédure Max (→X : numérique, →Y : numérique, ↔ M : numérique)
```

```
Début
```

```
    A ← X
```

```
    Si A < Y alors
```

```
        A ← Y
```

```
    Fsi
```

```
    M ← A
```

```
Fin // de la procédure
```

```
Début du programme principal
```

```
    Variables A, B, C : numériques
```

```

A ← 3
B ← 7
C ← 0
Max (A, B, C)
Ecrire « Le maximum de », A, « et », B, « est », C
Fin du programme principal

```

Exercice IV : Créer un petit ensemble de procédures et de fonctions permettant de manipuler facilement les heures et les minutes et composé de :

- 1- La fonction *Minutes*, qui calcule le nombre des minutes correspondant à un nombre d'heures et un nombre de minutes donnés.
- 2- La fonction ou la procédure HeuresMinutes qui réalise la transformation inverse de la fonction Minute.
- 3- La procédure AjouteTemps qui additionne deux couples de données heures et minutes en utilisant les deux fonctions précédentes.

Exercice V : Cet exercice permet de compléter les procédures et fonctions de l'exercice précédent

- 1- Créer une fonction qui permet de dire si un mois a 30 jours ou non. Cette fonction renverra 1 si c'est le cas et 0 sinon.
- 2- Créer une fonction qui permet de dire si un mois a 31 jours ou non. Cette fonction renverra 1 si c'est le cas et 0 sinon.
- 3- Créer une fonction qui permet de dire si une année est bissextile ou non. Cette fonction renverra 1 si c'est le cas et 0 sinon.
- 4- Utiliser ces fonctions pour écrire une fonction retournant le nombre de jours pour un mois et une année donnés.
- 5- Ecrire un programme principal permettant à l'utilisateur d'entrer un numéro de mois (entre 1 et 12) et une année (entre 1582 et 2003), qui seront ensuite passés en paramètres à la fonction **Nombre_de_jours()**.