

# TP1

## **Exercice 1:**

Écrire un algorithme qui permet de calculer le carré d'un nombre entier.

## **Exercice 2:**

Écrire un algorithme qui permet de calculer  $F(x, y)$  tel que  $x$  et  $y$  sont des nombres réels et  $F(x, y) = (x + y^2)/5$ .

## **Exercice 3:**

Lancer l'environnement PLATO et écris le programme suivant qui permet de faire l'addition de deux nombres:

```
program addition
implicit none
real::answer,x,y
print*,'Enter two numbers'
read*,x
read*,y
answer=x+y
print*,'the total is:',answer
end program addition
```

- Lancer la compilation (Project->Compile).
- Lancer l'édition de liens (Project->Build).
- Lancer l'exécution (Project->Run).
- Voir les fichiers: «.f90», «.obj», «.exe».

## **Exercice 4**

Lancer l'environnement PLATO et écris le code suivant :

```
program bug
Ce programme contient des erreurs
real :: a,b,ca = b + cread *,cprint *,a
end program simple
```

- Lancer le compilateur
- Corriger les erreurs
- Exécuter le programme

## **Traduction d'algorithmes en programme Fortran:**

Le tableau suivant contient les instructions Fortran équivalentes aux mots clés utilisés dans les algorithmes :

<u>Algorithme</u>	PROGRAM
Entier	INTEGER
Réel	REAL
←	=
Lire	READ *,
Ecrire	PRINT *,
Fin	END PROGRAM

- a) A l'aide de ce tableau traduisez les algorithmes des exercices 1 et 2 en programmes Fortran et sauvegardez ces derniers dans des fichiers sources .f90 ou f95, puis lancez la compilation, l'édition de liens, et l'exécution de chaque programme.

### **(Exercices facultatifs)**

#### **Exercice 5:**

On dispose de trois variables A, B et C. Ecrivez un algorithme transférant à B la valeur de A, à C la valeur de B et à A la valeur de C (toujours quels que soient les contenus préalables de ces variables).

**Traduire cet algorithme en fortran.**

#### **Exercice 6**

Ecrire un algorithme qui convertit en Octets, Kilo octets, Mega octets et Giga octets un nombre donnée en bits.

**Traduire cet algorithme en fortran.**

#### **Exercice 7**

Ecrire un algorithme qui lit au clavier l'heure et les minutes et qui affiche l'heure qu'il sera une minute plus tard. Par exemple, si l'utilisateur tape 21, puis 30, l'algorithme doit répondre : "Dans une minute, il sera 21 heure(s), 31 minute(s) ".

**Traduire cet algorithme en fortran.**