

## TP6 : Entrées/Sorties

Vous devez à la fin de la séance savoir compiler "à la main" et utiliser un makefile pour compiler. Vous trouverez un exemple de makefile à l'adresse suivante : <http://www.math.u-bordeaux1.fr/~dobj/f90/makefile>

### 1 Ecran et clavier

**Exercice 1 :** Le but de cet exercice est d'écrire un programme qui affiche de différentes manières les valeurs d'une matrice carrée  $2 \times 2$ .

1. Dans un fichier `libre.f90`, créer une subroutine permettant d'afficher à l'écran une matrice carrée  $2 \times 2$  en format libre.
2. Dans un fichier `float.f90`, créer une subroutine permettant d'afficher à l'écran une matrice carrée  $2 \times 2$  avec des nombres flottants comportant 2 chiffres après la virgule et 4 chiffres en tout,
3. Dans un fichier `sci.f90`, créer une subroutine permettant d'afficher à l'écran une matrice carrée  $2 \times 2$ , en notation scientifique avec 3 chiffres après la virgule.
4. Dans un fichier `prog.f90`, créer un programme qui appelle les 3 subroutines précédentes et testez-le avec les matrices ci-dessous

$$\begin{pmatrix} 3.444 & 2.34 \\ 10.555 & 89.9 \end{pmatrix}, \quad \begin{pmatrix} 378.444 & 29.34 \\ 100.555 & 89122.9 \end{pmatrix}, \quad \begin{pmatrix} 3 & 2 \\ 100000 & 1 \end{pmatrix}.$$

**Exercice 2 :** Ecrivez un programme qui affiche à l'écran le vecteur suivant :

$$( 1 \ 20 \ 300 \ 40000 \ 50000 \ 60 \ 7 \ 8 \ 9 \ 10 )$$

Pour cela, vous définirez et utiliserez un format qui affiche 10 fois un entier composé de 5 chiffres suivis de 2 espaces.

### 2 Fichiers

**Exercice 3 :** Ecrire un programme qui :

1. lit au clavier les différentes valeurs d'une matrice carrée  $2 \times 2$ ,
2. écrit dans un fichier non formaté les informations suivantes :
  - l'ordre de la matrice,
  - les valeurs de la matrice.

**Exercice 4 :** Ecrire un programme qui :

1. lit dans un fichier non formaté (créé par le programme de l'exercice précédent) les informations suivantes :
  - l'ordre de la matrice,
  - les valeurs de la matrice,
2. affiche ces informations à l'écran.

**Exercice 5 :** Ecrire un programme qui :

1. lit au clavier les différentes valeurs d'une matrice carrée  $2 \times 2$ ,
2. écrit dans un fichier formaté les informations suivantes :
  - l'ordre de la matrice,
  - les valeurs de la matrice.

**Exercice 6 :** Ecrire un programme qui :

1. lit dans un fichier formaté les informations suivantes :
  - l'ordre de la matrice,
  - les valeurs de la matrice,
2. affiche ces informations à l'écran.

**Exercice 7 :** Définissez un format qui permette 4 fois l'affichage d'une chaîne de caractère de 40 caractères suivie d'un retour à la ligne.

Utilisez ce format pour afficher à l'écran le texte suivant<sup>1</sup> :

Sables

Il n'est pas de desert si vaste

Que ne puisse traverser

Celui que porte la musique des etoiles

### 3 Modules

**Exercice 8 : OBLIGATOIRE** Cet exercice fait suite à l'exercice 5 du TP4. On représente toujours le polynôme  $P(x) = \sum_{i=0}^n a_i X^i$  par un type dérivé **polynome** composé :

- d'un entier **degre** qui représente le degré du polynôme
- d'un tableau de réels **coefficients** à une dimension contenant 101 cases numérotées de 0 à 100, représentant les coefficients du polynôme (on suppose que le degré du polynôme est au maximum 100).

---

<sup>1</sup>Ce poème a été écrit par M. LE SAINT qui a remporté un prix lors du 5e concours de poésie RATP.

Exemple : Pour  $P$ , l'entier `degre` vaut  $n$  et le tableau `coefficients` est

$$(a_0, a_1, \dots, a_n, 0, \dots, 0)$$

Dans un fichier `modpoly.f90`, créer un module contenant :

1. le type dérivé `polynome`.
2. la fonction `initialize_pol`
3. la subroutine `print_pol`
4. la subroutine `cree_monome`
5. la fonction `copie_pol`
6. la fonction `add_pol`
7. la subroutine `mult_pol`
8. la fonction `derivpol`

Dans un fichier `poly.f90`, faites appel à ce module et tester que tout marche correctement.

1. Ecrivez sur papier l'algorithme effectuant la division euclidienne de deux polynômes
2. En se servant des sous-programmes écrits dans les questions précédentes, toujours dans le module `modpoly`, écrire une subroutine `divisioneucl` qui effectue la division euclidienne de 2 polynômes. Cette subroutine prend comme paramètres :
  - deux polynômes,
  - le reste de la division,
  - le quotient de la division.

## 4 Corriger des programmes

**Exercice 8 :** Les programmes suivants ne compilent pas ou plantent lors de leur exécution. Corrigez ces programmes pour qu'ils compilent et tournent correctement :

1. <http://www.math.u-bordeaux1.fr/~dobj/f90/exo9-1.f90>  
Pour exécuter ce programme, il faut que vous récupériez le fichier suivant :  
<http://www.math.u-bordeaux1.fr/~dobj/f90/fichier1>
2. <http://www.math.u-bordeaux1.fr/~dobj/f90/exo9-2.f90>
3. <http://www.math.u-bordeaux1.fr/~dobj/f90/exo9-3.f90>

## 5 Encore des exercices sur les fichiers et les formats

**Exercice 7 :** Le but de cet exercice est de se familiariser avec des programmes qui lisent et écrivent des fichiers.

1. Voici un premier programme :

<http://www.math.u-bordeaux1.fr/~dobj/f90/exformatte.f90>

Télécharger-le ainsi que le fichier : <http://www.math.u-bordeaux1.fr/~dobj/f90/bat.amdba>

Dans ce programme, on veut lire un fichier dont le nom est donné par l'utilisateur et écrire dans un fichier nommé `toto`. Lisez le programme, compilez-le (en corrigeant bien sur les différentes erreurs qui apparaissent) et faites le tourner deux fois. Tout se passe-t-il bien ? Pourquoi ?

2. Télécharger le programme suivant : <http://www.math.u-bordeaux1.fr/~dobj/f90/exerreur.f90> ainsi que les fichiers : <http://www.math.u-bordeaux1.fr/~dobj/f90/ficherr>

<http://www.math.u-bordeaux1.fr/~dobj/f90/bat.amdba>

Dans ce programme, on veut gérer correctement les erreurs. Assurez-vous que vous comprenez tout ce qu'il y a écrit dans le programme, compilez et exécutez-le avec les fichiers téléchargés. Que se passe-t-il ? Pourquoi ?