Travaux Pratiques d'algorithmique n°1

Cours d'Informatique de Deuxième Année

—DEUG Sciences S3-MI—

Pointeurs et Tableaux

L'objet de cette séance de travaux pratiques est le rappel des concepts de base du langage C, concepts normalement déjà assimilés pour la plupart d'entre vous. Nous rappelons entre autres : les tests, les boucles, les fonctions et le passage des paramètres.

Deux notions importantes du langage C seront introduites en fin de séance :

- les pointeurs ;
- l'allocation dynamique d'espace mémoire.

Toutes les fonctions programmées devront être commentées.

▸ **Exercice 1. (factorielle d'un entier)** *La fonction factorielle est définie par les relations suivantes :*

$$n! = \begin{cases} 1 & \text{si } n = 0 \ , \\ n \times (n-1) \times \cdots \times 3 \times 2 \times 1 & \text{si } n > 0 \ . \end{cases}$$

Ecrire la fonction `Factorielle` *qui prend en argument* `n` *et qui calcule la factorielle de* `n`.

▸ **Exercice 2. (coefficients binomiaux)** *Nous cherchons à calculer les coefficients binomiaux C_n^p définis par les relations suivantes :*

$$C_n^p = \begin{cases} \dfrac{n!}{p!(n-p)!} & \text{si } 0 \leq p \leq n \ , \\ \\ 0 & \text{sinon.} \end{cases}$$

Ecrire la fonction `Binomial` *qui prend comme arguments* `n` *et* `p` *et qui calcule le coefficient binomial C_n^p.*

▸ **Exercice 3.** *Nous cherchons à étudier les propriétés des coefficients binomiaux. Nous nous intéressons à la somme suivante :*

$$\sum_{i=0}^{n} C_n^i \quad .$$

Ecrire la fonction `Somme` *qui prend en argument* `n` *et qui calcule cette somme. Essayer cette fonction pour $n = 2, 3, 4, 5, \ldots$. En déduire une évaluation de la somme ci-dessus.*

▸ **Exercice 4.** *Ecrire une fonction* `SommeEntier` *qui prend en argument un entier et qui calcule la somme des chiffres composant cet entier.*

▸ **Exercice 5.** *Soit le programme C suivant :*

```
#include <stdio.h>
void main(void)
{ int a, b, *c;
  a=2; b=a; c=&a;
  printf("%d %d %d\n",a,b,*c);
  a=3;
  printf("%d %d %d\n",a,b,*c);
}
```

Commentez l'affichage du programme et rajoutez une instruction qui affiche l'adresse de `a` *et la valeur de* `c`*.*

▸ **Exercice 6.**

- *Ecrire une fonction permettant de saisir un tableau de réels. Le prototype de la fonction sera* `double * SaisirTableau(int *size)` *où* `size` *est la taille du tableau ;*
- *écrire une fonction* `void MinMax(double *T, int size, int *min, int *max)` *qui recherche les indices des cases du minimum et du maximum dans le tableau* `T` *de taille* `size`*;*
- *écrire une fonction* `Moyenne` *qui retourne la moyenne arithmétique des valeurs contenues dans un tableau* `T` *de taille* `size`*;*
- *écrire une fonction* `Norme` *qui retourne la norme d'un tableau* `T` *(considéré comme un vecteur) de taille* `size`*.*

▸ **Exercice 7.** *A l'aide de l'instruction* `switch`*, créer un menu permettant d'utiliser toutes les fonctions du TP.*
Exemple de menu :

```
switch( variable ){
  case val_1 : instructions; break;
  case val_2 : instructions; break;
  .... .....   ............  .....
  case val_k : instructions; break;
  default    : instructions_par_defaut;
}
```