Travaux Pratiques de Programmation C n°1

Cours d'Informatique de Deuxième Année

—L2—

Quelques rappels sur le langage C

Durant cette séance de travaux pratiques, nous faisons le rappel des concepts de base du langage C (tests, boucles, tableaux, fonctions, ...). Nous verrons aussi un concept très important à appréhender qui est la **différence entre le passage par valeur et par adresse des arguments**.
Afin de prendre de bonnes habitudes (en vue notamment du projet), **l'ensemble des fonctions programmées devront être commentées et testées**. Nous rappelons aussi que les options appliquées lors de la compilation seront **-Wall** et **-ansi** et qu'il est **très fortement déconseillé d'utiliser des variables globales**.

▸ **Exercice 1.** ***(Passage par valeur ou par adresse)***
Programmer et tester les trois fonctions suivantes :

```
void fct1(int a, int b) {
  int tmp;
  tmp = a;
  a = b;
  b = tmp;
}
```

```
void fct2(int a, int *b) {
  int tmp;
  tmp = a;
  a = *b;
  *b = tmp;
}
```

```
void fct3(int *a, int *b) {
  int tmp;
  tmp = *a;
  *a = *b;
  *b = tmp;
}
```

Laquelle de ces fonctions permet d'échanger les valeurs de deux variables ? Justifier votre réponse.

▸ **Exercice 2.** ***Nombre factoriels***
On dit qu'un entier strictement positif x est un ***nombre factoriel*** *s'il existe un autre entier y tel que $x = y\,!$.*

- *Écrire une fonction qui, étant donné un entier x, vérifie si x est un nombre factoriel et, si c'est le cas, calcule la valeur de y tel que $x = y\,!$.*
- *En déduire un programme complet.*

▸ **Exercice 3.** ***Nombres parfaits***

Un ***nombre parfait*** *est un nombre (entier positif) présentant la particularité d'être égal à la somme de tous ses diviseurs, excepté lui-même. Ainsi par exemple, 6 est un nombre parfait (6 = 1 + 2 + 3).*

- *Écrire une fonction* `est_parfait` *qui prend en argument un entier positif n et qui retourne 1 si n est parfait et 0 dans le cas contraire.*
- *Ecrire une fonction qui affiche tous les nombres parfaits inférieurs à un entier n.*
- *Ecrire une autre fonction qui affiche les n premiers nombres parfaits.*
- *Enfin, écrivez un programme complet permettant de tester toutes ces fonctions. Pour ce faire, vous utiliserez un "menu" proposant à l'utilisateur les différents choix possibles.*

▸ **Exercice 4.** ***Manipulation de tableaux de caractères***

Dans la suite les tableaux peuvent contenir au plus MAXLETTRES(=50) caractères. Écrire les fonctions pour :

- *lire un tableau de caractères ; la fonction renvoie le nombre de caractères lus et arrête la saisie lorsque le tableau est plein ou que l'utilisateur a appuyé la touche <<ENTREE>>,*
- *afficher sur une ligne un tableau de taille caractères ; l'affichage du tableau sera suivi sur la ligne suivante par celui des codes ASCII des caractères,*
- *compter les voyelles dans un tableau de taille caractères,*
- *compter les caractères qui ne sont ni des chiffres ni des lettres dans un tableau de taille caractères,*
- *remplacer dans un tableau de taille caractères, chaque voyelle en minuscule par la voyelle en majuscule correspondante.*

Il est conseillé d'écrire des fonctions additionnelles :

`int EstChiffre(char c)` *qui renvoie 1 si* `c` *est un chiffre,*

`int EstVoyelle(char c)`,

`char MinusMajus(char c)` *qui renvoie la majuscule correspondante si* `c` *est une minuscule et renvoie* `c` *sinon.*