

# Algorithmique et Programmation

## TP3

### Manipulation de grands entiers

## 1 Introduction

On cherche à effectuer des calculs sur des nombres entiers comportant un nombre de chiffres plus important que les types entiers standard. Chaque entier sera représenté par l'ensemble de ses chiffres (en base 10), le nombre de chiffres étant limité par la constante `MAXCHIFFRES`. On construit la structure de données suivante :

```
type
  t_Chiffres : vecteur d'entiers
  t_EntierLong : enregistrement
    booléen : négatif
    t_Chiffres : chiffres
  fin_enregistrement
```

Pour un entier  $x$ , **négatif** vaut *vrai* si le nombre est négatif. Les valeurs du tableau `chiffres` sont les valeurs des  $(a_i) \in \{0..9\}$  (rangés de gauche à droite) tels que

$$|x| = \sum_{i=0}^{MAXCHIFFRES-1} a_i 10^i$$

Ainsi, le nombre 357635735 sera représenté dans le tableau `chiffres` de la façon suivante :

indice	0	1	2	3	4	5	6	7	8	9	...	19
chiffre	5	3	7	5	3	6	7	5	3	0	...	0

**Traduction C :** Vous trouverez sur le serveur pédagogique le fichier `entierlong.h` qui contient la définition du type structuré `t_EntierLong`. On rappelle que les tableaux en C commencent à l'indice 0.

## 2 Affichage et saisie d'un entier long

### 2.1 Définitions des fonctions

```
Spécification
Fonction ecrireEntierLong(n)
Paramètre : t_EntierLong n
Résultat : aucun // l'entier long n sera affiché à l'écran
```

**Spécification****Fonction**  $n \leftarrow \text{lireEntierLong}()$ **Paramètre** : aucun // un entier long sera saisi au clavier (avec un nombre de chiffres inférieur ou égal à MAXCHIFFRES)**Résultat** :  $t\_EntierLong\ n$  // l'entier long saisi au clavier sera rangé dans  $n$ 

## 2.2 Traduction en C

Ces deux fonctions ont déjà été écrites pour vous en C dans le fichier `lit_ecrit.c`. Pour pouvoir utiliser ces fonctions, vous disposez, sur le serveur pédagogique, des fichiers suivants :

- `lit_ecrit.h` : contient les prototypes des fonctions fournies,
- `lit_ecrit.o` : fichier objet issu de la compilation du fichier `lit_ecrit.c`

## 3 Travail à réaliser (lisez jusqu'au bout)

Concevez et programmez en C les fonctions suivantes :

- Dans un fichier `utilitaires.c` :
  - conversion d'un entier standard en entier long ;
  - test d'égalité de deux entiers longs ;
  - comparaison en valeur absolue : pour deux entiers longs  $n1$  et  $n2$ , a-t-on  $|n1| \leq |n2|$  ?
- Dans un fichier `operations.c` :
  - addition  $n1 + n2$  de deux entiers longs  $n1$  et  $n2$  **de même signe** ;
  - soustraction  $n1 - n2$  de deux entiers longs  $n1$  et  $n2$  de même signe et tels que  $|n1| \geq |n2|$  ;
  - addition et soustraction de deux entiers longs de signes quelconques en faisant appel aux fonctions précédentes ;
  - *Pour les plus rapides* : division euclidienne de deux entiers longs positifs.

**Simultanément**, dans un fichier `main.c`, écrivez une fonction principale que vous pourrez modifier progressivement pour tester au fur et à mesure les différentes fonctions programmées.

## 4 Compte-rendu

À la fin des 4h du TP, vous rendrez un compte-rendu contenant les éléments suivants :

- Pour chaque fonction :
  - la spécification ;
  - une description brève du principe mis en œuvre ;
  - des jeux d'essais simples illustrant son bon fonctionnement ;
  - le programme C.
- L'algorithme de la fonction addition de même signe ;
- Les jeux d'essais correspondant au calcul de la suite de Fibonacci :

$$u_0 = 0, u_1 = 1, \forall n \geq 2, u_n = u_{n-1} + u_{n-2}.$$

Pour plusieurs valeurs de  $n$ , dont une très élevée de l'ordre de 20 000, calculer les  $n$  premiers termes de la suite, afficher les 3 derniers.