

# Licence 1 - section B

## TD 7 d'éléments d'informatique

Catherine RECANATI – Département d'Informatique – Institut Galilée

Semaine du 28 novembre au 2 décembre 2016

*Pour produire le code source des programmes : écrire d'abord l'algorithme (en mélangeant français et instructions C), puis écrire le programme, sous forme de commentaires. N'écrivez pas vos commentaires après coup ! Le but est de remplir le squelette de programme que vous avez écrit avec des commentaires, pour aboutir au texte du programme. Il ne s'agit pas ici de commenter les lignes de votre programme, mais de vous permettre de les produire.*

### 1 Tableaux et fonctions à argument de type tableau

ATTENTION : dans tous ces énoncés, nb représente le nombre d'éléments du tableau tab passé en premier argument à la fonction.

#### Exercice 1.1 Initialisations de tableaux de int ou de char

1. Ecrire une procédure `void initTabInt(int tab[], int nb, int val);` qui initialise un tableau de variables de type `int`, comportant nb éléments, uniformément avec la valeur `val`.
2. Même question pour la procédure `initTabChar`, qui initialise un tableau de caractères uniformément avec une valeur de type `char` nommée `val`.

#### Exercice 1.2 Comparaisons de tableaux d'entiers

1. Définir le type énuméré `booléen` comme constitué des deux valeurs constantes `VRAI` et `FAUX`, elles mêmes définies avec `#define` par les constantes `1` et `0`.
2. Ecrire une fonction `booléen compare(int tab1[], int tab2[], int nb);` qui retourne `VRAI` si les deux tableaux sont égaux (i.e. s'ils ont les mêmes valeurs), et qui retourne `FAUX` dans le cas contraire.
3. Modifier la fonction précédente pour écrire une fonction `int different(int tab1[], int tab2[], int nb);` qui retourne le rang du premier élément qui diffère dans les deux tableaux (de même taille), et le nombre zéro si les deux tableaux n'ont que des valeurs identiques.

#### Exercice 1.3 Recherche d'une valeur dans un tableau

1. Ecrire une fonction `int searchValue(int val, int tab[], int nb);` qui recherche la présence d'une valeur `val` dans le tableau `tab` de nb éléments. La fonction retourne l'indice de la première variable du tableau qui a cette valeur, et `-1` si le tableau ne contient pas cette valeur.
2. Modifier la fonction précédente pour qu'elle puisse s'appliquer à un tableau de caractères.

#### Exercice 1.4 Tableaux de char

1. `void readTabChar(int tab[], int nb);`  
Cette procédure initialise un tableau de nb caractères avec une série de caractères qui sont lus au clavier avec `scanf`.
2. `void imprimeChar(char tab[], int nb);`  
Cette procédure imprime un tableau de nb caractères en affichant successivement chacun de ses éléments (sans passage à la ligne suivante).
3. `void traduire(short tab[], int nb, char ascii[]);`  
Cette procédure effectue la traduction d'un tableau d'entiers (`short`) dans un tableau de caractères de même taille nommé `ascii[]` dans lequel pour chaque `i < nb`, `ascii[i]` sera le caractère (type `char`) dont le code ASCII est `tab[i]`.

**Exercice 1.5** Imprimer un tableau de caractères sous la forme [a b z d ...].

1. De manière analogue à ce qui a été vu en cours avec la procédure `imprime` qui prenait en argument un tableau de variables de type `int`, écrire une autre procédure `imprime` qui prend en arguments un tableau de caractères et le nombre d'éléments de ce tableau, puis affiche entre deux crochets les caractères du tableau un à un en les séparant par des caractères espace. Si le tableau passé en argument est vide, on affichera simplement deux crochets ouvrant et fermant [ ].
2. Si vous avez écrit cette fonction en utilisant une boucle `while`, transformez le code source pour utiliser une boucle `for` (et inversement).