

TP 11

Les fichiers de structures, à organisation séquentielle

On considère des fichiers représentant des annuaires téléphoniques, où chaque abonné est de type *abonne* défini comme suit :

```
typedef char t_chaine50[51] ;
typedef char t_chaine10[11] ;

typedef struct{
    t_chaine50 c_nom;        // nom de l'abonné
    t_chaine10 c_num;        // numéro de cet abonné
    int c_duree;             // durée pour laquelle il est abonné
} t_abonne ;
```

Dans chacune des procédures manipulant le fichier des abonnés, le nom du fichier physique sera passé en paramètre ; le fichier logique et la zone tampon seront des variables locales.

Exemple :

```
void lister (char nomFic[] ) {        // liste le contenu du fichier de
                                     //nom nomFic
    FILE * f ;
    t_abonne a;
    ...
}
```

Question 1 : Écrire un programme permettant à l'utilisateur, via un menu :

- soit de créer un fichier contenant les renseignements sur les abonnés, chaque enregistrement comportant le nom, le numéro, et la durée (nombre de mois) de l'abonnement d'un abonné. Les informations seront lues au clavier (la fin des données est signalée par "*" à la place du nom de l'abonné),
- soit de lister ce fichier à l'écran,
- soit d'ajouter de nouveaux abonnés à la fin du fichier.

Question 2 : Ajoutez une nouvelle option à votre menu qui permet de trier le fichier des abonnés sur leur nom de manière croissante. Soit le type *t_tabAbo* défini comme suit :

```
#define MAX 100
typedef t_abonne t_tabAbo[MAX] ;
```

Pour trier le fichier vous procéderez en trois étapes :

1. recopie du fichier dans un tableau de type *t_tabAbo*,
2. tri du tableau,
3. recopie du tableau dans le fichier.

Écrivez les procédures et fonctions suivantes :

- `int copierFichierDansTableau(t_chaine50 nomFichier, t_tabAbo t)` qui insère les abonnés du fichier dans le tableau *t* et qui retourne le nombre d'abonnés recopiés (on supposera qu'il est inférieur à *MAX*).
- `void maintientTrie(t_tabAbo t, int i)` { ... } qui fait en sorte que *t[0..i]* soit trié sachant que *t[0..i-1]* est trié, en faisant "remonter" *t[i]* par permutation (cf tp 10 exercice 2). Cette procédure utilisera la procédure `void permuter(t_abonne* a1, t_abonne* a2)` qui permute les abonnés **a1* et **a2*.
- `void copierTableauDansFichier(char nomFic[], t_tabAbo t, int nb)` qui écrit chaque élément du tableau *t* dans le fichier

Options supplémentaires :

Compléter le programme avec les options suivantes (cf td 11):

- pouvoir afficher à l'écran le numéro d'un abonné dont on fournit le nom.
- pouvoir prolonger l'abonnement de tous les abonnés (la durée de prolongation sera saisie au clavier),
- pouvoir prolonger l'abonnement d'un abonné dont on fournit le nom (le nom de l'abonné et la durée de prolongation seront saisis au clavier) ,
- pouvoir supprimer un abonné dont le nom est saisi au clavier.