

## Fonctions – Chaînes de caractères – Fichiers

3 séances

### Gestionnaire de contacts

Il s'agit d'écrire un programme en langage C qui permet d'entrer, de sauvegarder dans un fichier, et de rechercher des contacts dans un répertoire. Le squelette du programme est fourni. Les fichiers **rep.h** et **rep.c** sont complets et ne doivent pas être modifiés.

Vous allez fabriquer deux modes de fonctionnements pour ce programme. L'un utilise un **tableau**, l'autre une **liste chaînée** pour stocker les contacts.

Le fichier **rep.h** contient une directive qui permet de choisir le mode de fonctionnement du programme :

```
#define IMPL_TAB // si défini, on compile pour une implémentation tableau du
répertoire
// #define IMPL_LIST // si défini, on compile pour une implémentation Liste Chaînée du
répertoire
```

Dans un premier temps, vous devez mettre au point la version **tableau**. La version **liste chaînée** sera réalisable quand cette notion aura été abordée en cours.

Seules les fonctions du fichier **outil.c** doivent alors être complétées.

Les fichiers **sources** doivent être déposés sur Campus dans une archive (prenom-nom.zip) à l'issue des trois premières séances.

Une première version du programme doit utiliser un **tableau** pour stocker les données en mémoire.

La version complète du projet comportera les **deux implémentations, tableau et liste chaînée** dans le fichier **outil.c**.

Les informations à prendre en compte pour chaque personne sont le nom, le prénom et un numéro de téléphone.

Les contacts sont sauvegardés dans un fichier au format texte dans le même répertoire que le programme et nommé **rep.txt**. Les lignes du fichier sont au format : **nom;prénom;0320304050**

L'application **rep** peut être appelée avec un paramètre qui précise un nom différent pour le fichier de données **rep.txt**.

Exemple : `rep perso.txt`

#### Evolution :

Les fonctions seront ensuite modifiées pour utiliser une liste chaînée. Le projet devra alors intégrer deux nouveaux fichiers ( **liste.h** et **liste.c** ) fournissant les fonctions nécessaires à la manipulations d'une **liste chaînée unilatère** dans laquelle les enregistrements seront toujours triés.

## Les écrans de l'application

### Menu Principal :

Repertoire :

(A)fficher tout

a(J)outer une personne

recherche(R)e une personne

e(X)it

L'option **eXit** permet de quitter le programme. Si une modification a été apportée aux contacts, une sauvegarde est proposée avec possibilité de choisir un nom de fichier. Si aucun nom n'est saisi, c'est le nom par défaut **rep.txt** qui est proposé.

### Menu Afficher Tout :

Adageron,	Olivier 03-20-33-99-55
Claudine,	François 03-20-12-30-17
Courte,	Eric 03-20-62-76-70
_____ entree/espace/x ____ :	

Remarque :

**entree** affiche une ligne à la fois.

**espace** affiche une page à la fois lorsque les lignes ne tiennent pas sur une page.

**x** permet de revenir au menu principal.

### Menu Recherche

Recherche :

par (N)om

par (T)elephone

e(X)it

### Sous-Menu Rechercher

Nom ? :dupont

Dupont, Pascal

03-20-44-33-89

\_\_\_\_\_/espace/(D)elete/e(X)it \_\_\_\_  
:

L'enregistrement trouvé peut être supprimé sans confirmation.

Une recherche infructueuse renvoie le message « n'existe pas dans le répertoire ».

La touche espace permet d'afficher les occurrences suivantes si elles existent.

La recherche par no de téléphone permet de saisir des espaces ou des tirets sans que cela ne perturbe la recherche.

La recherche des noms ne respecte pas la casse minuscules-majuscules.

### Menu ajouter une personne

Nom : Legrand

Prenom :Theo

Telephone :03 20 30 40 50

Il est possible de ne pas renseigner le champ prénom ou téléphone.

Une fois le contact créé on revient automatiquement au menu principal.

### Réalisation du TP :

Le projet Visual Studio est constitué d'un fichier principal source **rep.c** qui contient le programme principal, d'un fichier d'entêtes **rep.h** et d'un fichier **outil.c** qui contient les fonctions à compléter par

vos soins. Il faut donc créer un projet avec ces différents éléments afin de pouvoir compiler le programme.

Dans **outil.c**, le nom des fonctions proposées, les définitions globales NE DOIVENT PAS ETRE MODIFIES.

Vous ne devez pas créer de nouvelles fonctions, hormis pour la mise en place d'une liste chaînée dans les fichiers **liste.c** et **liste.h**, pour la suppression d'un élément notamment.

Ecrivez le code des fonctions sans ajouter de variable globale.

Un commentaire en début de fichier précise votre nom et votre prénom.

Le fichier d'entêtes **rep.h** permet de choisir une implémentation tableau ou une implémentation liste chaînée à l'aide des `#define IMPL_TAB` ou `#define IMPL_LIST`. Il faut mettre en commentaire l'option que vous ne souhaitez pas conserver.

La manipulation des fichiers se fera avec les fonctions suivantes du langage C:

- `FILE *fic_rep` : déclaration d'une variable de type fichier
- `err = fopen_s(&Fic_rep, char * nom_fichier, "w")` : ouverture d'un fichier en écriture-remplacement  
<https://docs.microsoft.com/fr-fr/cpp/c-runtime-library/reference/fopen-s-wfopen-s?view=vs-2017>
- `fclose(FILE *fic_rep)` : fermeture du fichier
- `fgets(char * buffer, long_max_rec, int fic_rep)` : lecture d'une ligne dans le fichier
- `feof(fic_rep)` : renvoie VRAI si la fin de fichier est atteinte
- `fputs(char * buffer, int fic_rep)` : écriture d'une ligne dans le fichier

L'écriture formatée de chaînes de caractères dans un buffer peut se faire à l'aide de la fonction `sprintf_s()`.

**REMARQUE** : Les opérations de lectures et écritures sont réalisées en une seule fois au lancement et en sortie de programme par les fonctions **charger()** et **sauvegarder()**. Toutes les autres opérations sont réalisées en mémoire, soit sur le tableau, soit sur la liste chaînée.

Si vous utilisez des noms de contact comportant des caractères accentués, il faut lancer votre exécutable dans une console CMD.exe après avoir passé la commande **CHCP 1252** ou vérifier que cette commande est passée par un appel système dans le programme C.