

SDA1: TP6-7

Le répertoire téléphonique

Aucune variable GLOBALE !!

Toute variable doit être déclarée au sein d'une séquence { }

L'objectif du programme est la gestion d'un **répertoire** téléphonique. Les éléments de ce répertoire (aussi nommé agenda) sont des **fiches** (contenant chacune une personne dont on connaît le **nom**, l'**adresse**, le **numéro de téléphone**, la **ville** et le **code postal**).

Soit la structure de données suivante:

```
# define CAPACITE_REPERTOIRE 10
# define MAX 4

# define K_Maxpersonnes CAPACITE_REPERTOIRE+1
// +1 pour la sentinelle (case 0)
// les fiches seront rangées entre la case 1 et la case 10

# define K_MaxNom MAX
# define K_MaxAdr MAX
# define K_MaxVil MAX
# define K_MaxCode MAX
# define K_MaxTel MAX

typedef char T_Nom[K_MaxNom] ;
typedef char T_Adr[K_MaxAdr];
typedef char T_Vil[K_MaxVil];
typedef char T_Tel[K_MaxTel];
typedef char T_Code[K_MaxCode];

typedef struct
{
    T_Nom Cnom ;
    T_Adr Cadr ;
    T_Vil Cvil ;
    T_Code Ccode ;
    T_Tel Ctel ;
} T_Fiche;

typedef T_Fiche T_Tableau[K_Maxpersonnes] ;
```

```
typedef struct
{
    T_Tableau Cpersonnes;
    int Ctaille ;
    //Ctaille permet de mémoriser le nombre de fiche(s) contenue(s) par
    // Crepertoire
} T_Repertoire ;
```

```
void main(int argc, char *argv[])
{
T_Repertoire rep ;

// ici le menu

    system("PAUSE");
}
```

*Inutile de TOUT saisir !!
Cette structure de données est sur moodle.*

IMPORTANT

Le type T_Tableau est un tableau de 11 T_Fiche mais la première fiche sera rangée à la case n°1.

La case 0 sera utilisée pour la technique de la sentinelle (utile pour des recherches diverses ou autres tris ...)

Ecrire un programme de gestion de répertoire en intégrant dans ce programme les fonctionnalités suivantes (testez au fur et à mesure) dont vous préciserez la signature (prototype):

Un menu sera proposé au sein du main

Partie 1

Sans gérer l'unicité du nom dans le répertoire

Sous-programme lire_nom

Saisie d'une chaîne de caractères et « renvoie » de cette chaîne.

Sous-programme lire_fiche

Saisie du contenu d'une fiche

appel à lire_nom mais aussi et sur le même modèle à **lire_adr**, **lire_tel** ...

Sous-programme afficher_fiche

Affiche le contenu de la fiche à l'écran.

Sous-programme ajouter_fiche

Introduire la fiche dans le répertoire (si répertoire non complet, incrémenter Ctaille)

Sous-programme lister_repertoire

Affiche à l'écran les noms des personnes présentes dans le répertoire.

Sous-programme editer_repertoire

Affiche à l'écran la totalité du répertoire. Les infos d'une personne se trouvent sur 2 lignes.

Sous-programme recherche_ville :

Saisie du nom de la ville ;
Donne le nom des personnes habitant dans cette ville.

Sous-programme recherche_nom:

Saisie du nom de la personne ;
Donne le telephone des personnes portant ce nom

Partie 2**En gérant l'unicité du nom dans le répertoire****« Pas de doublon sur le nom »****Sous-programme appartient**

Recherche le nom dans le répertoire . Utilisation de la sentinelle.
Renvoie un résultat booléen

Sous-programme modif_fiche

Saisie le nom de la personne ;
Vérifie l'existence;
Ecrit le contenu de sa fiche à l'écran ;
Saisie des nouvelles informations (adresse, ville, tel, code postal).

Sous-programme consulter_fiche

Saisie le nom de la personne ;
Vérifie l'existence ;
Ecrit le contenu de sa fiche à l'écran ;

Sous-programme recherche_nom

Saisie du nom ;

Vérification de l'existence ;
Indique si la personne existe ou non.

Sous-programme recherche_fiche

Saisie du nom et du contenu total de la fiche
Indique si cette fiche est présente dans l'agenda

Sous-programme suppression

Saisie le nom de la personne ;
Vérifie l'existence ;
Suppression de la fiche