

TP 7 : Caractères et chaînes de caractères

Gestion d'un agenda téléphonique (2 séances)

On souhaite gérer un répertoire téléphonique contenant des noms (chaînes de caractères) et des numéros de téléphone (entiers). Le programme pourra lire un nom et un numéro pour l'ajouter au répertoire, afficher tout le répertoire, rechercher un nom dans le répertoire et afficher son numéro.

Vous procéderez par étapes en validant chacune d'elles au fur et à mesure de la progression.

Partie I

On définira les variables globales suivantes :

```
#define MAXcar 20
#define MAXnom 100
typedef char typeNom[MAXcar+1] ; // une case supplémentaire pour '\0'
typeNom tableNom[MAXnom] ; // table des noms
long tableTel[MAXnom] ; // table des numéros
int nbPersonne=0 ; // nbre de personnes enregistrées
```

tableNom		tableTel	
dupont	0	0491202112	<div style="display: flex; align-items: center;"> <div style="border-left: 1px solid black; border-right: 1px solid black; height: 100px; margin-right: 5px;"></div> <div style="writing-mode: vertical-rl; transform: rotate(180deg);">ordre de saisie</div> </div>
bonaventure	1	0491202113	
bonarien	2	0491202132	
zoe	3	0491202912	
	4		
	5		
	6		

1. Lecture d'un nom (construction d'une chaîne)

Ecrivez et testez la fonction `lireNom` qui prend en argument une variable de type `typeNom`, lit une suite de caractères terminée par la touche <entrée> ('`\n`' ou code ASCII égal à 10) et la stocke dans cette variable après les modifications suivantes :

- Les premiers caractères séparateurs seront ignorés* (commencez la fonction `lireNom` par `scanf(" ")`)
- Les blancs intermédiaires seront ignorés
- Les majuscules seront transformées en minuscules,
- Les noms dépassant `MAXcar` caractères seront tronqués
- La variable respectera le format d'une chaîne de caractères (suite de caractères terminée par '`\0`')

Par exemple, la saisie de "Du Pont" doit être remplacée par la chaîne "dupont"

d	u	p	o	n	t	\0			
---	---	---	---	---	---	----	--	--	--

(*) Rappel : Dans la question suivante, vous devrez lire un nom suivi d'un numéro de téléphone pour enrichir votre agenda. L'alternance de lectures de caractères et de nombres pose le problème de la touche <entrée> qui est à la fois un caractère et/ou la validation d'une ligne de saisie. La saisie d'un nombre par une instruction du type `scanf("%d", &i)` sera terminée sans être lu par le caractère <entrée> qui restera dans le buffer de lecture et qui peut « polluer » la prochaine lecture d'un caractère ou d'une chaîne de caractères.

La solution pour traiter la touche <entrée> uniquement comme une validation consiste à ajouter un blanc (et un seul) dans le format de lecture :

- soit en fin du format de lecture d'un nombre : `scanf("%d ", &i);`
- soit en précédant la lecture d'un caractère ou d'une chaîne par l'instruction `scanf(" ");`

2. Menu pour gérer l'agenda

En utilisant les structures de données définies au début de ce TP, écrivez le programme qui propose par menu :

0. Quitter l'application
1. Ajouter une personne
2. Afficher tout l'agenda

L'ajout d'une personne consiste à

- lire un nom (utilisez `lireNom`) et un numéro de téléphone
- les ajouter à la suite de ceux déjà insérés dans les tableaux `tableNom` et `tableTel`
- incrémenter le nombre de personnes enregistrées

Afficher l'agenda consiste à lister la totalité des noms et leurs numéros de téléphone dans l'ordre de saisie.

3. Rechercher

Ecrivez la fonction `rechercher` qui

- reçoit un nom en paramètre
- recherche séquentiellement le rang de ce nom dans `tableNom`
- retourne ce rang ou -1 si le nom est absent

Complétez le menu pour avoir la fonctionnalité ci-dessous qui lit un nom au clavier (utilisez `lireNom`), appelle la fonction `rechercher` avec ce nom en paramètre et affiche ce nom et son numéro de téléphone s'il existe.

3. Rechercher

Partie II

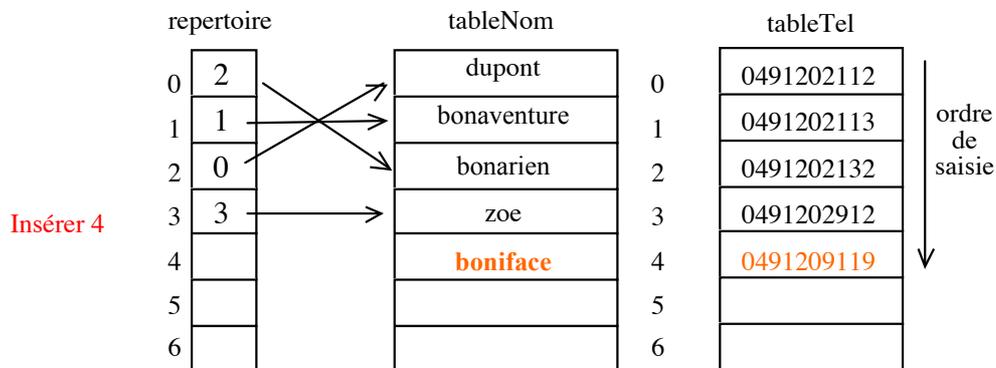
On souhaite maintenant obtenir un affichage du répertoire dans l'ordre alphabétique. Aussi, vous ajouterez la variable globale ci-dessous qui sera chargée de mémoriser les indices des noms dans l'ordre alphabétique.

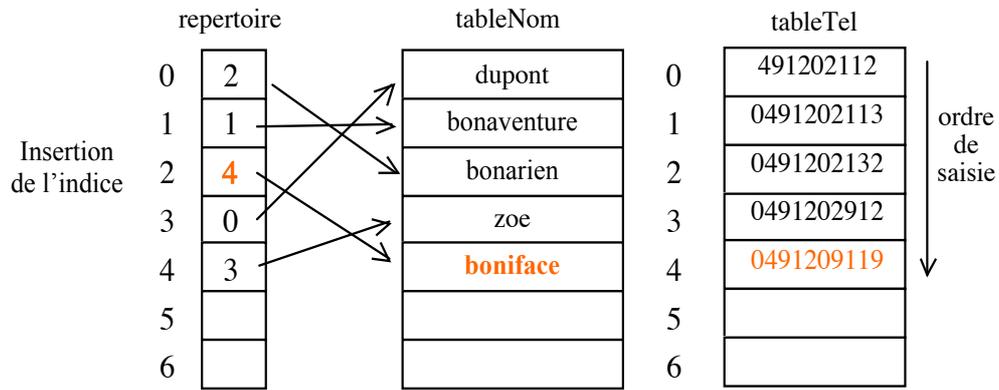
```
int repertoire[MAXnom] ; /* classement des noms dans l'ordre alpha. */
```

1. Modification de la fonction ajout :

Les tableaux `tableNom` et `tel` contiennent les noms et numéros de téléphone dans l'ordre où ils ont été saisis. Lors d'une nouvelle saisie, le nouveau nom et son numéro de téléphone sont stockés à la suite des noms et numéros déjà enregistrés.

Toutefois, pour bénéficier d'un classement alphabétique, le tableau `repertoire` sera mis à jour au fur et à mesure des saisies de sorte que son parcours séquentiel donne les indices des noms suivant l'ordre alphabétique. L'ajout d'un nouvel indice dans le tableau `repertoire` se fera donc de préférence avec un "tri par insertion".





2. Affichage :

Réécrivez la fonction affichage pour éditer les noms avec leurs numéros dans l'ordre alphabétique.