

TD 1(Les fichiers)

Exercice 1

Ecrire un programme C qui permet de créer et relire un fichier texte nommé "suite.txt" de 10 premiers entiers. La suite présentée sous la forme suivante:

12345678910

Exercice 2

Écrire un programme qui crée un fichier texte nommé "tableX.txt" qui contienne la table de multiplication pour le nombre X (le fichier "table7.txt" contiendra la table de 7), présentée sous la forme suivante :

1 x 7 = 7

2 x 7 = 14

...

10 x 7 = 70

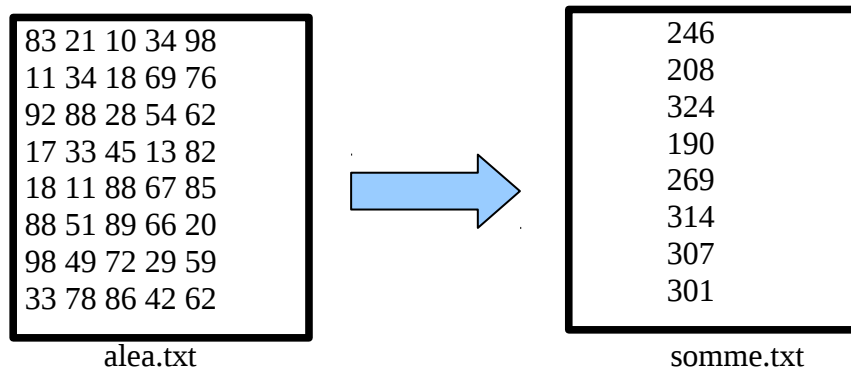
Exercice 3

Question 1. Écrire la fonction d'entête **void creer_alea(char source[])** qui permet de créer un fichier source contenant un nombre aléatoire de lignes (entre 4 et 10 lignes). De plus chaque ligne contient cinq valeurs entières qui sont aléatoires.

Question 2. Écrire la fonction d'entête **void SommeLigneFich(char *Fs, char *Fd)**. Cette fonction permet de remplacer chaque ligne du fichier texte de nom physique Fsrc (le fichier Fs en premier paramètre est supposé existant) par la somme de ces nombre. Les résultats de chaque somme du fichier Fs seront mises dans un nouveau fichier texte de nom physique Fd (2ème paramètre) sera crée dans la fonction)

Exemple d'exécution

>>>SommeLigneFichier('alea.txt','somme.txt')



Exercice 3

Question 1. Écrire la fonction d'entête **void MinToMajLigne(char src[], char dest[])** qui retourne une nouvelle chaîne de caractères en transformant tous les caractères de la chaîne src en majuscule.

Question 2. Écrire la fonction d'entête **void MinToMajFichier(char fichMin[], char fichMaj[])**. Cette fonction permet de transformer en majuscule toutes les lignes du fichier texte de nom physique fichMin (le fichier fichMin en premier

paramètre est supposé existant). Les lignes transformées en majuscule du fichier fichMin seront mises dans un nouveau fichier texte de nom physique fichMaj.

Exercice 4

Q1. Créer sur le disque dur puis afficher à l'écran le fichier INFORM.TXT(fichier texte) dont les informations sont structurées de la manière suivante:

- nom (20 caractères maximum) ;
- Age (entier) ;
- note (réel)

Les informations relatives aux différents étudiants (N) seront lues au clavier et affichées sur l'écran .

Q2. On reprend le même travail en utilisant un fichier binaire.

Exercice 5

Un marchand d'automobiles d'occasion souhaite gérer son stock de 30 voitures. Chaque voiture est caractérisée par les informations suivantes:

- Sa marque
 - Modèle
 - Prix
 - Numéro d'immatriculation
1. Ecrire une fonction **creer_fichier** qui prend en paramètre un flot fich de type FILE * et le nom du fichier à créer, et ouvre le fichier écriture. Ensuite écrire dans ce fichier les informations de cet marchand automobiles.
 2. Ecrire une fonction **lire_fichier** qui prend en en paramètre un flot fich de type FILE * et le nom du fichier ouvert en lecture, et permet de lister toutes les informations de chaque voiture.
 3. Dans **la fonction main**, utiliser les deux fonctions creer_fichier et lire_fichier

Exercice 6

Ecrire un programme qui détermine dans un fichier un texte dont le nom est entré au clavier:

- le nombre de caractères qu'il contient,
- le nombre de mots,
- le nombre de paragraphes (c.-à-d.: des retours à la ligne),

Les retours à la ligne ne devront pas être comptabilisés dans les caractères. On admettra que deux mots sont toujours séparés par un ou plusieurs des caractères suivants:

- fin de ligne
- espace
- ponctuation: . : , ; ? !
- parenthèses: ()
- guillemets: "
- apostrophe: '

Utiliser une fonction d'aide SEPA qui décide si un caractère transmis comme paramètre est l'un des séparateurs mentionnés ci-dessus. SEPA restituera la valeur (logique) 1 si le caractère est un séparateur et 0 dans le cas contraire. SEPA utilise un tableau qui contient les séparateurs à détecter.