



TD/TP N°2 : Gestion de fichier

Exercice 1 :

Soit un fichier de données, « data.txt » structuré en une suite de lignes contenant chacune un nom de personne, un nom d'article, un nombre d'article et un prix unitaire pour chaque article.

Exemple :

Ali Cartable 10 100

Youness Tonner 5 350

Ecrire la fonction main() dans laquelle on déclare les variables suivantes :

- nom et article : tableaux de 80 caractères,
- nombre, prix_unitaire : de type entier,

Le corps de la procédure consistera en une boucle dont chaque itération lira une ligne et l'imprime à l'écran. La lecture se fera par un appel à fscanf() affectant les 4 champs de la ligne aux 4 variables nom, article, nombre et prix_unitaire. L'écriture consistera à imprimer nom, article et nombre*prix_unitaire.

Exercice 2 :

Parti 1 :

Écrivez une fonction compte_c(FILE *f) qui renvoie le nombre de caractères d'un fichier.

Écrivez une fonction compte_m(FILE *f) qui renvoie le nombre de mots d'un fichier. Les mots sont séparés par des espaces ou des retours à la ligne.

Écrivez une fonction compte_l(FILE *f) qui renvoie le nombre de lignes d'un fichier.

Parti 2 :

Faites un programme qui prend en argument sur la ligne de commande le nom d'un fichier et une lettre (c, m, ou l). Selon la lettre, le programme doit renvoyer soit le nombre de caractères, soit le nombre de mots, soit le nombre de lignes du fichier.

Exercice 3 :

Vous allez créer un programme qui gère une base de données simple d'étudiants. Chaque étudiant a un nom, un prénom, un âge et une note finale. Les données des étudiants seront stockées dans un fichier binaire.

Tâches à Réaliser :

1. Créez une structure Etudiant contenant les champs suivants :
 - o char nom[50]
 - o char prenom[50]
 - o int age
 - o float note

2. Écrivez les fonctions suivantes :

- void ajouter_etudiant(FILE *fichier): Cette fonction demande à l'utilisateur d'entrer les informations d'un étudiant et les ajoute à la fin du fichier.
- void afficher_etudiants(FILE *fichier): Cette fonction lit tous les étudiants du fichier et affiche leurs informations à l'écran.
- void chercher_etudiant(FILE *fichier, char *nom): Cette fonction recherche un étudiant par son nom et affiche ses informations. Si l'étudiant n'est pas trouvé, un message approprié doit être affiché.

3. **Fonction main () :**

- Créez un menu qui permet à l'utilisateur de choisir une option :
 - Ajouter un étudiant
 - Afficher tous les étudiants
 - Chercher un étudiant par nom
 - Quitter le programme

4. Assurez-vous de gérer les erreurs lors de l'ouverture et de la lecture du fichier.