

Durée : 3 séances - Dossier à rendre pour la semaine du 11/03/19.

Il s'agit de réaliser un petit outil permettant la gestion de messages à partir d'une liste chaînée. La structure de chaque message est de la forme :

- Date de début de validité du message (gérée **en entier** sous la forme aaaammjj)
- Date de fin de validité du message (même forme),
- Texte du message (maximum 100 caractères),
- Pointeur vers le message suivant.

Les données permettant de créer la liste chaînée sont lues dans un fichier, à raison d'une ligne par message.

Important : la liste chaînée est triée sur la date de début. Dans le fichier d'entrée les messages ne sont pas triés.

Le fichier est sans erreur.

1) Créer un module de gestion de listes chaînées contenant les opérations élémentaires nécessaires pour gérer cette liste chaînée (création de cellule, recherche du précédent, insertion d'une cellule dans un premier temps). Ces opérations seront écrites conformément aux principes vus en cours (par exemple la bonne gestion d'un précédent).

Utiliser les opérations de ce module pour créer la liste chaînée des messages à partir du fichier d'entrée.

Vérifier la liste créée grâce au debugger (ddd).

Important : Ne pas faire de menu, le nom du fichier doit être en argument de la ligne de commande.

2) Écrire une fonction qui sauvegarde la liste dans un fichier de même structure que le fichier d'entrée.

Écrire une fonction qui affiche les messages non expirés à la date du jour. La date du jour sera obtenue grâce à la date système (cf. polycop de C).

3) Écrire une fonction qui supprime les messages devenus obsolètes (le gestionnaire de la liste chaînée sera donc complété par une opération élémentaire de suppression d'une cellule).

Écrire une fonction qui change la date de début de messages connaissant leur date de début initiale (tous les messages ayant cette date de début initiale seront traités).

4) Question (bonus pour les plus rapides). Ecrire une fonction qui affiche les messages contenant un motif (chaîne de caractères) donné (quelle que soit les dates).

CONDITION D'ACCES AUX MACHINES

Chaque séance de TP doit être préparée, les algorithmes doivent être écrits et seront vérifiés par les enseignants.

TRAVAIL A FAIRE :

A la fin de chaque séance envoyer à l'enseignant le code produit.

REDIGER LE DOSSIER DE PROGRAMMATION CORRESPONDANT :

Présentation générale

- une description de l'objet du TP (3 lignes),
- une description et un schéma de la structure de données et des fichiers de données utilisés,
- l'organisation du code source (dans quel fichier source sont les fonctions).

Détail de chaque fonction (programme principal compris)

- les algorithmes de principe (dans l'entête),
- le lexique des notations (paramètres et variables locales dans l'entête),
- les programmes commentés (respecter le guide de style),

Compte rendu d'exécution

- **le make file**
- des jeux de test complets (données en entrée et résultats obtenus par copie d'écran et de fichier), **tous les cas particuliers seront testés** (en faire la liste).