

ISIMA 1^{ère} ANNEE- 2012-2013
TP1 de Structure de données – langage C

Durée : 4 séances - Dossier à rendre pour le 22/03/2013.

On gère un échéancier (ou un agenda) grâce à une liste chaînée à deux niveaux. Chaque bloc de la liste du premier niveau est de la forme :

- année sur 4 caractères,
- numéro de semaine (caractères de 01 à 53) – année et numéro peuvent être gérés dans le même champ.
- Pointeur sur la liste des actions de la semaine,
- Pointeur vers la semaine suivante.

Chaque bloc de la liste chaînée des actions (second niveau) est de la forme :

- jour de la semaine (de 1 à 7)
- heure (de 00 à 24) – jour et heure peuvent être gérées dans le même champ.
- nom de l'action (champ fixe de 10 caractères non terminé par NULL)
- Pointeur vers l'action suivante.

La liste des semaines est triée en croissant sur le n° de semaine, seule les semaines non vides sont présentes. La liste des actions est triée en croissant sur le jour et l'heure.

1) Créer une structure de données initiale en mémoire à partir d'un **fichier texte, où chaque ligne donne, sans séparateur :**

- année, semaine, jour, heure (sans espace),
- libellé de l'action,

Exemple de ligne : **201321110TP de SDD**

On utilisera 2 fonctions d'insertion (l'une pour la semaine, l'autre pour l'action) indépendantes du traitement du fichier. Ces fonctions devront utiliser exactement les algorithmes écrits en cours (précédent en pointeur de pointeur). **Le fichier est supposé trié et sans erreur. Les listes sont triées dans le même ordre que le fichier.**

Remarque : Ne pas faire de menu ou le faire le plus simple et le plus court possible. Par exemple : ne pas mettre la lecture du fichier entrée au menu, le nom de ce fichier doit être en argument de la ligne de commande.

2) Écrire une fonction qui sauvegarde la SDD dans un fichier de même structure qu'en 1). Écrire une fonction qui crée une liste contiguë des jours où une action contenant un motif (chaîne de caractères) donné est présente.

3) Écrire les fonctions de recherche et de suppression d'une action connaissant année, semaine, jour, heure (séparer les fonctions sur la semaine de celles sur l'action). Ces fonctions devront utiliser exactement les algorithmes écrits en cours (précédent en pointeur de pointeur).

4) Transformer la liste des semaines en liste bilatère.

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 :

1 question = 1 séance. 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,
- le lexique des notations (paramètres et variables locales en 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).

Le TP noté sur le contenu du dossier, la **qualité et la modularité du code** et la **complétude** des jeux de tests