

## **P0001-3 : Structures simples de fichiers -- Méthodes d'index**

### **PROBLÈME : Transformation de structures de fichiers**

On définit les structures de fichiers suivantes :

**T Ô V  $\bar{C}$**  : Tableau non ordonné, longueur variable des articles, non chevauchement des articles.

**T O V  $\bar{C}$**  : Tableau ordonné, longueur variable des articles, non chevauchement des articles.

**T O F** : Tableau ordonné, longueur fixe des articles.

1. Construire à partir de clés générées aléatoirement, un fichier **T Ô V  $\bar{C}$** . Un article est composé d'une clé sur 4 caractères et d'une chaîne de caractères de longueur variable positive. On tolérera les doubles.

2. Transformer le fichier **T Ô V C** en un fichier **T O V C** de la manière suivante :

a) Construire à partir du fichier **T Ô V C** une table triée selon la clé en mémoire contenant toutes les clés avec les adresses vers les articles correspondants. Le tri se fera au fur et à mesure des insertions.

b) Créer le fichier **T O V C**, à raison d'au plus &% par bloc (& donné), à partir de cette table et du fichier **T Ô V C**.

3. On veut doter le fichier obtenu **T O V C** avec un fichier d'index **T O F index** contenant uniquement les dernières clés des blocs. Un article du fichier d'index est le couple (Clé, n° de bloc).

a) Construire le fichier d'index **T O F** à partir du fichier **T O V C**.

b) Ecrire le module de chargement de l'index.

c) Ecrire le module de recherche d'un article de clé donné. On procèdera par dichotomie.

### **CONSEILS**

Les fichiers de données peuvent être définis comme suit :

**Fdon UN FICHIER DE ( ENTIER , VECTEUR ( 50 ) DE CAR ) BUFFER**  
**Bufdon ;**

L'entier désigne le nombre d'articles dans le bloc.

A l'intérieur d'un bloc, les articles sont séparés par '/ '.

Le fichier d'index peut être défini comme suit :

**Findex UN FICHIER DE ( ENTIER , VECTEUR ( 5 ) DE CHAINES ,**  
**VECTEUR ( 5 ) DE ENTIERS ) BUFFER Bufindex ;**

L'entier désigne le nombre de paires ( clé, adresse), Le premier vecteur les clés, le second vecteur les adresses vers les articles correspondants.

L'expression **ALEACHAINE ( 4 + 1 + ALEANOMBRE ( 31 ) )** permet de générer un article composé d'une clé sur 4 caractères et d'une chaîne de longueur minimale 1. **ALEANOMBRE ( 31 )** fournit un nombre aléatoire entre 0 et 30.

### **MACHINE ABSTRAITE SUR LES FICHIERS**

**OUVRIR (Fl, Fp, Mode) :**Ouvrir le fichier logique Fl et l'associer au fichier physique Fp en précisant le mode( fichier nouveau('N') ou ancien 'A') )

**FERMER (Fl) :** Fermer le fichier Fl.

**LIRESEQ (Fl, V) :** Lire dans la variable tampon V le bloc (ou l'article)se trouvant à la position courante.

**ECRIRESEQ (Fl, V):** Ecrire le contenu de la variable tampon V à la position courante du fichier Fl.

**LIREDIR (Fl, V, N) :** Lire le N-ième bloc (ou article) du fichier Fl dans la variable tampon V.

**ECRIREDIR (Fl, V, N) :** Ecrire le contenu de la variable tampon V à la N-ième position du fichier Fl.

**RAJOUTER(Fl, V) :** Ecrire le contenu de la variable tampon à la fin du fichier Fl.

**FINFICH(Fl) :** Prédicat égal à vrai si la fin du fichier Fl est rencontrée, faux sinon.

**ALLOC\_BLOC(Fl) :** fournit un bloc (ou article) du fichier dans lequel on pourra écrire.

**ENTETE(Fl, I) :** Récupérer la I-ième caractéristique du fichier Fl.

**AFF\_ENTETE(Fl, I, Exp) :** Affecter Exp comme la I-ème caractéristique du fichier.