

SAE - Guide : lecture/écriture des fichiers .lex

Les indications données ici n'ont pas valeur de consignes, ce sont des pistes pour structurer votre travail. Vous ne serez pas pénalisés si vous choisissez de ne pas suivre ce guide, mais ce sera peut-être plus difficile de vous aider si vous vous en éloignez trop.

Conseils généraux

- Toutes les fonctions manipulant le fichier binaire sont dans une librairie (.h/.c) faisant appel aux autres bibliothèques sur les arbres lexicographiques
- Avoir une structure représentant l'en-tête d'un fichier
- La structure pour chaque cellule est a priori la même que pour un arbre binaire statique en mémoire
- La fonction d'écriture (arbre en mémoire -> fichier) est relativement simple, il s'agit d'écrire l'en-tête puis le contenu du tableau en séquentiel
- Il n'y a pas besoin de la fonction inverse (fichier -> arbre en mémoire). A la place, il faut au moins
 - une fonction permettant d'ouvrir le fichier et de lire l'en-tête
 - une fonction lisant une cellule d'indice donné (prenant aussi un pointeur vers le fichier ouvert et les informations d'en-tête en paramètre)
- Pour la lecture, il est conseillé d'avoir une structure contenant
 - l'en-tête du fichier
 - le pointeur vers le fichier ouvert (type FILE*) C'est cette structure qui servira de type "arbre lexicographique dans un fichier" dans les autres fonctions.
- Vous pouvez ensuite écrire des fonctions plus avancées (recherche de caractère, de mot, ...)

Lecture/Ecriture dans un fichier binaire

Voir par exemple ici pour des tutoriels:

[<https://www.programiz.com/c-programming/c-file-input-output#read-write-binary>]

On utilisera:

- **fopen** et **fclose** pour ouvrir/fermer le fichier binaire
- **fwrite** pour l'en-tête ou tout le tableau de cellules dans le fichier
- **fseek** pour chercher une position spécifique dans le fichier
- **fread** pour lire l'en-tête ou une cellule (après appel à fseek)