

TD 6 de Programmation Bas Niveau : malloc/realloc, fichiers

Aurélien BOSSARD

Pré-requis : chapitres 1 à 9 du cours de Programmation Bas Niveau.

Exercice 1

Écrivez les fonctions suivantes :

- **int afficheFichier (char * chemin)**; qui affiche en entier sur la sortie standard le fichier dont le nom est chemin, renvoie 1 si la lecture s'est déroulée correctement et -1 si une erreur est survenue. Utilisez la fonction fread.
- **char * litLigne (FILE *fp)**; qui lit la ligne courante du fichier représenté par le curseur fp caractère par caractère, et la copie dans une chaîne de caractères allouée dynamiquement à chaque lecture, qui sera renvoyée. Utilisez la fonction fread.
- **char ** litFichier (char * chemin, int * nbLignes)**; qui lit le fichier chemin ligne par ligne et renvoie un tableau de caractères qui contient sur la première dimension les lignes, et sur la deuxième chaque caractère de chaque ligne. La fonction met à jour nbLignes avec le nombre de lignes lues (et donc allouées au tableau). Le tableau doit être alloué dynamiquement au fur et à mesure de la lecture du fichier.
- **int ecritFichier (char * chemin, char ** contenu, int nbLignes)**; qui effectue l'opération inverse de la fonction précédente. La fonction doit renvoyer -1 si une erreur est survenue. Utilisez la fonction fwrite.
- **void libereTab (char ** tab, int nbLignes)**; qui libère le tableau tab qui contient nbLignes et qui a été alloué dynamiquement avec litFichier.

Exercice 2

On veut stocker et charger un fichier de listePoint2D. Écrire les fonctions suivantes :

- **int stocke (char * chemin, ListePoint2D * lp)**; qui stocke chacun des points de la liste lp dans le fichier chemin. Attention, il ne faut pas stocker le pointeur vers l'élément suivant, il ne servirait à rien. Il faut donc uniquement enregistrer les deux membres abscisse et ordonnée de chaque point.
- **int charge (char * chemin, ListePoint2D * lp)**; qui charge dans lp chacun des points enregistrés dans le fichier chemin. Il faut donc charger des float deux par deux, dans le même ordre qu'ils ont été sauvegardés, et ajouter un nouvel élément à lp pour chaque couple lu.