

COMPTE RENDU - TP N°1

Structures de données – ZZ1

Le 20/03/2020

Table des matières

I.	Objet du TP	2
a)	Exemples des Structure des données manipulées :	2
II.	Description des fichiers en entrée	3
III.	Organisation du code source	3
IV.	MakeFile	3
V.	Jeu de test.....	3

I. Objet du TP

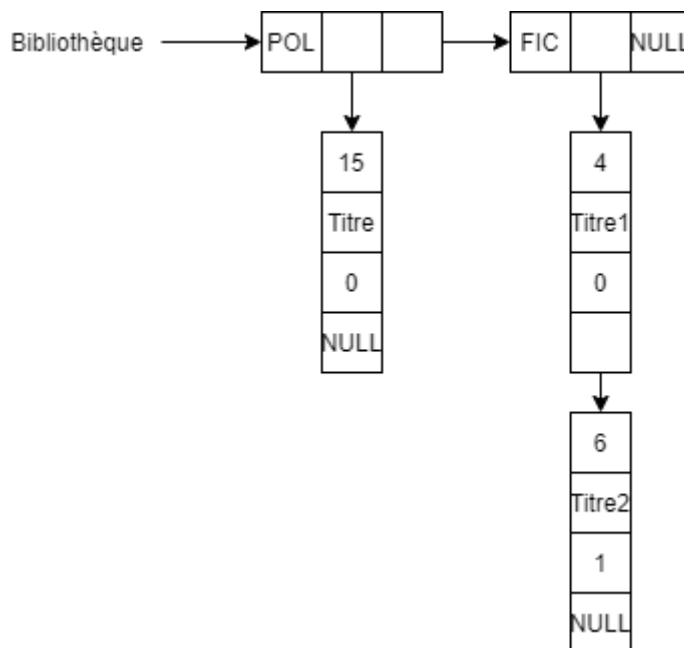
L'objet de ce TP consiste à manipuler des listes chaînées. Pour ce faire nous devons mettre en place 2 structures principales.

La première liste chaînée constitue notre bibliothèque, c'est une liste chaînée à deux niveaux : la première cellule contient le nom d'une catégorie, la 2^{ème} est un pointeur vers une liste chaînée de nos livres et la 3^{ème} cellule contient un pointeur vers la catégorie suivante. La liste chaînée des livres dispose de 4 cellules, la première contient le numéro du livre, la 2^{ème} le titre du livre, la 3^{ème} un booléen indiquant si le livre est emprunté et le 4^{ème} contient un pointeur vers le livre suivant. La liste des livres est rangée par ordre croissant des numéros des livres.

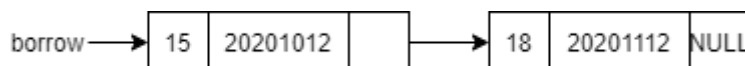
La deuxième liste chaînée contient 3 cellules : une cellule pour le numéro du livre emprunté, une autre pour la date de remise et une dernière qui pointe sur le prochain emprunt. Cette liste est triée par ordre croissant sur les dates de retour.

a) Exemples des Structure des données manipulées :

Structure de la bibliothèque :



Structure des emprunts :



II. Description des fichiers en entrée

Nous avons trois types de fichier texte en entrée, un pour ajouter des livres dans notre bibliothèque, un autre pour ajouter des emprunts et un autres pour supprimer des emprunts.

Le format des fichiers pour ajouter des livres a déjà été défini dans la consigne :

POL 3	1ère ligne : catégorie, nombre de livres dans cette catégorie
11 le tueur	2ème ligne : n° de livre, titre
15 la victime	
17 le truand	
BD 2	
21 tintin	
23 titeuf	

Pour le format des fichiers ajout des emprunts nous avons choisi le format ci-dessous :

1 20211532	1 ^{ère} colonne : numéro du livre.
18 20200116	2 ^{ème} colonne : date de retour du livre.

Enfin le format des fichiers suppression des emprunts nous avons choisi le format ci-dessous :

18	1 ^{ère} colonne : numéro du livre.
25	

III. Organisation du code source

Src/borrow : contient les fonctions qui gèrent directement la liste chaînée des emprunts

Src/lib : contient les fonctions qui gèrent directement la liste chaînée de la bibliothèque

Borrow_file : contient les fonctions pour gérer les emprunts depuis un fichier extérieur

lib_file : contient les fonctions pour gérer la bibliothèque depuis un fichier extérieur

main : permet de tester toutes nos fonctions.

IV. MakeFile

Notre Makefile se trouve dans notre dossier au chemin bibli/Makefile

V. Jeu de test

3 fichiers de texte pour les tests dans bibli/data/. Ces trois fichiers insèrent en début, milieu et fin de chaîne.

Add_borrow.txt : pour tester les ajouts d'emprunts.

Lib.txt : pour tester l'ajout des livres.

Remove_borrow.txt : pour tester les suppression d'emprunts.

Notre jeu de test se trouve dans notre main.c (bibli/src/main.c) Il S'appelle avec 3 arguments :

- 1 - Le chemin pour le fichier de lib (data/lib.txt)
- 2 - Le chemin pour le fichier de borrow (data/borrow.txt)
- 3 - Le chemin pour le fichier de add_borrow (data/add_borrow.txt)
- 4 - Le chemin pour le fichier de remove_borrow (data/remove_borrow.txt)

Soit la commande suivante : « prog data/lib.txt data/borrow.txt data/add_borrow.txt data/remove_borrow.txt »

Nos fonctions addBorrowFromFile, removeBorrowFromFile, loadLib gèrent ou non les cas suivants :

AddBorrowFromFile gère le cas où le livre n'existe pas et le cas où le livre est déjà emprunté.

RemoveBorrowFromFile gère le cas où le livre n'existe pas mais ne gère pas le cas où l'emprunt n'existe pas.

Loadlib ne gère pas les cas où le fichier ne respecte pas le format exemple :

2 livres annoncés dans la catégorie BD mais le nombre de livre qui suit n'est pas respecté

BD 2

21 tintin

23 titeuf

41 les schtroumpfs