
Travail synthèse – Bibliothèque de la MRC Saint-Clin-de-banlieue

420-C21-VM Programmation II

Remise le **26 mai 2022 à 23h59**

Directive : La remise sera faite sur Léa.

Objectif du projet

- Démontrer l'acquisition des compétences visées par les cours 420-C11-VM et 420-C21-VM.
- Vous mettrez en pratique tout ce que vous connaissez.

Description du problème

Ce travail se divise en deux sections. Dans la première partie, vous devez créer le nouveau logiciel de gestion de la Bibliothèque de la MRC de Saint-Clin-de-banlieue. Ce logiciel devra générer la liste des clients, des livres et des prêts. Pour y parvenir, votre application utilisera directement des fichiers binaires et sans maximum de client.

Les dates

Dans ce travail, votre application devra fournir différentes dates. Vous devez utiliser la fonction **Aujourd'hui** pour l'obtenir. Elle est disponible dans le fichier **utils.cpp**. De plus, vous ne pouvez pas demander de date dans la fonction **Retour**. La date de retour doit être envoyée à la fonction via ses paramètres.

Les actions

Vous devez créer un programme de gestion de la bibliothèque. Voici le menu principal du programme de gestion :

1. Nouveau livre
2. Nouveau client
3. Dossier du client
4. Location de livre
5. Retour des livres
6. Liste des livres prêtés
7. Liste des clients en retard
8. Quitter

La sélection d'un numéro de livre ou de client doit être faite dans la fonction main et elle doit être transmise aux fonctions en paramètre. Le numéro du livre ou du client correspond à leur position dans leur fichier de donnée.

Fonctions à utiliser

Voici la description des fonctions que vous devez utiliser. Attention ! Vous devez créer vos propres prototypes. La liste suivante décrit les noms et les objectifs des fonctions. De plus, vous pouvez en développer plus.



NouveauLivre

Cette fonction lit, au clavier, toutes les informations sur le livre. Il faudra aussi initialiser le prêt à faux. On n'affiche pas le numéro du livre dans cette fonction. L'ordre de lecture doit être de la forme suivante :

1. Titre du livre
2. L'auteur ou l'autrice du livre

La fonction doit converser ce livre dans le fichier *Livres.bin*.

NouveauClient

Cette fonction lit, au clavier, toutes les informations sur le client. Il faudra aussi initialiser le nombre de prêts à zéro. On n'affiche pas le numéro du client dans cette fonction. L'ordre de lecture doit être de la forme suivante :

1. Nom du client
2. Prénom du client
3. Téléphone
4. Numéro civique du client
5. Rue du client
6. Ville du client
7. Code postal du client (format X0X0X0 – aucun espace)

La fonction doit converser ce client dans le fichier *Clients.bin*.

RechercherLivre

À partir du numéro du livre (qui commence à 0), la fonction doit retrouver les informations du livre présent dans le fichier. Elle ne doit pas afficher ou demander d'information.

RechercherDossierClient

À partir du numéro de la personne cliente (qui commence à 0), la fonction doit retrouver le dossier présent dans le fichier. Elle ne doit pas afficher ou demander d'information.

AfficherLivre

À partir du numéro du livre (qui commence à 0), la fonction doit afficher toutes les informations du livre. Elle ne doit pas demander d'information.

AfficherDossierClient

À partir du numéro de la personne cliente (qui commence à 0), la fonction doit afficher toutes les informations du client. Toutefois, elle n'affiche pas la liste des livres empruntés. Elle ne doit pas demander d'information.

MettreAJourLivre

À partir du numéro du livre (qui commence à 0) et des nouvelles données du livre, la fonction doit modifier le fichier de donnée pour conserver les nouvelles informations.

MettreAJourClient

À partir du numéro de la personne cliente (qui commence à 0) et des nouvelles données de la personne cliente, la fonction doit modifier le fichier de donnée pour conserver les nouvelles informations.

Location

Une location comprend le numéro, la date courante et la date de retour prévu. Une personne cliente ne peut pas emprunter plus de trois livres. La date devra être fournie par une fonction **Aujourd'hui**. La location est pour 15 jours. Utilisez la fonction **AjouterJours** pour y ajouter un certain nombre de jours à une date. Elle doit modifier le livre et le client, dans leurs fichiers, pour le signifier comme prêté. Elle ne doit pas afficher ou demander d'information. La fonction attend deux informations : le numéro du livre et le numéro de la personne cliente.

Retour

Cette fonction modifiera le nombre de livres prêtés (le mettra à zéro). Si le client est en retard, on retourne le montant à payer (5\$ par livre par jour de retard, plus taxes (TPS : 5% et TVQ : 9.975%). Vous pouvez calculer le nombre de jours grâce à la fonction **NombreJours**.

Avertissement :

Dans votre application (mis à part les tests), vous devez toujours appeler la fonction Retour avec la date d'aujourd'hui.

ListeDesLivresPretes

Cette fonction affiche les titres et numéros de tous les livres prêtés en ce moment.

ListeDesClientsEnRetard

Cette fonction affiche les nom, numéro de téléphone et date de retour (au format jj/mm/aaaa) de chaque client en retard. La date affichée concerne un seul livre et toujours le premier livre en retard.

Les constantes à utiliser

Vous devez, dans ce travail, utiliser les trois constantes suivantes :

Nom	Type	Valeur	Portée
FILENAME_LIVRE	string	"Livres.bin"	Publique
FILENAME_CLIENT	string	"Clients.bin"	Publique
MAX_CHAR	int	125	Privé au fichier .h

La dernière constante est utilisée pour les tableaux de char dans les structures. Je vous suggère d'utiliser la même approche que vous avez utilisée dans le dernier TP.



Structure à développer

Pour chaque client, on stocke les informations suivantes:

- Nom et prénom
- Téléphone (10 caractères)
- Adresse (numéro civique, rue, ville, code postal)
- Date d'inscription (par défaut, la date d'aujourd'hui)
- les informations sur les livres prêtés (3 max):
 - numéro du livre
 - date et heure de location (type `Date_s`)
 - date et heure du retour prévu (type `Date_s`)

Enfin, vous devez créer un registre de livre (disponible dans son propre fichier). La structure devra avoir cette forme :

- Titre du livre
- Auteur
- Est-ce qu'il est prêté ou non ?

Avertissement :

Vos clients et vos ne doivent pas être conservé dans une variable globale (OBLIGATOIRE).

La liste binaire

Vous devrez stocker les clients dans un fichier binaire contenant des structures `Client_s`, auxquelles vous accéderez grâce à l'accès aléatoire (accès direct). Il en va de même pour les livres. Consultez les exemples dans les notes de cours (fichier binaire et pages 127-130 du manuel).

- Vous devrez remplacer les strings de vos structures par des chaînes de caractères `C`
- Vous pouvez utiliser la fonction `LireChaineC()` pour lire ces chaînes.

Les fichiers utilisés doivent être nommés comme suit : `Livres.bin` et `Clients.bin`. Leurs noms sont évocateurs de leur contenu. Les deux fichiers doivent être dans le même répertoire que le projet.

Avertissement :

Utilisez les constantes globales pour vos noms de fichier (OBLIGATOIRE). De plus, vous ne devez pas utiliser de collection (tableau, liste, etc.) pour conserver les données.

Affichage dans la console et lecture du clavier

Aucune fonction ne doit lire le clavier ou écrire sur la console sauf si la description le mentionne. De plus, vous devez inclure le fichier `flux_io.h` au lieu de `iostream`. De plus, vous devez utiliser `ecrire` au lieu de `cout` et `lire` au lieu de `cin`.

De plus, toutes les chaînes de caractères doivent être lues grâce à la fonction `LireChaineC`.



Répartition multifichier

Vous devez utiliser les fichiers et le code de base fourni. Vous pouvez ajouter vos propres éléments, mais il est obligatoire de respecter la structure qui vous est fournie.

Vous devez compléter les trois définitions de précompilation dans le fichier **type.h** (**MATRICULE**, **NOM**, **PRENOM**).

Avertissement :

Si vous ne respectez pas le cadre fourni, vous aurez la note de 0 pour ce travail.

Processeur de compilation

Vous devez compiler votre code en 64 bits.

Remise

Vous devez remettre les trois fichiers suivants :

- Tous vos fichiers .h et .cpp
- Incluez les fichiers **cvm23.h** et **cvm23.cpp**

Avertissement :

Ne remettez pas votre exécutable.

La remise doit avoir lieu **suivant les modalités sur la page principale**. La remise doit avoir lieu sur **Léa uniquement**. La remise est le **26 mars 2023 à 23h59**. Les copies sont acceptées jusqu'au **5 avril 2023 à 15h30**.

Utilisation de tableau

Votre travail doit s'appuyer sur les fichiers binaires. Si vous utilisez des tableaux pour contenir la liste des livres et/ou des personnes clients au lieu des fichiers binaires, votre note ne pourra pas dépasser 50%.

Compilation du programme et modification des tests

Si votre programme ne compile pas, votre note ne dépassera pas 10%. Si vous modifiez les tests, vous obtenez la note de 0.

Critères d'évaluation

L'évaluation de votre code sera sur 50 points. Pour tous les travaux remis en retard, vous perdrez 10% par tranche de 24 heures de retard. Ce travail a une pondération de 20 % de la note finale.

Test du programme

Votre note est calculée par l'ensemble des points accordé par les tests de l'application.

Erreur de programmation ou autre

Chaque erreur entraîne une pénalité qui est réduite du résultat des tests.