

## 1 Cahier des charges

Ce devoir est à faire en binôme.

La date de remise du projet est :

— mardi 24 novembre 2020 à 18h00

Ensuite chaque groupe passera un oral de 20 minutes pour exposer son travail et répondre aux questions.

Le projet sera déposé sur la plate-forme UPDAGO, sous forme d'un fichier source IMPÉRATIVEMENT nommé *Collection.c* (notamment en respectant la casse). Ce fichier ne doit pas être compressé.

Le langage utilisé est obligatoirement le C. Votre fichier doit compiler, sans erreur ni warning, sous *gcc* avec les options suivantes :

-*Wall* -*Wextra* -*pedantic* -*std=c99* (voire -*Wconversion* si vous avez le courage).

De même un programme utilisant votre fichier doit s'exécuter avec *valgrind* sans erreur ni fuite mémoire.

Vous n'êtes pas autorisés à utiliser des bibliothèques ou des composants qui ne sont pas de votre cru, hormis les bibliothèques système. En cas de doute, demandez l'autorisation.

Il vous est demandé un travail précis. Il est inutile de faire plus que ce qui est demandé. Dans le meilleur des cas le surplus sera ignoré, et dans le pire des cas il sera sanctionné.

## 2 Conditions de développement

Le but du projet est de reprogrammer le TP 5 mais en utilisant une autre structure de données que les tableaux pour la collection de voitures.

Au vu de l'architecture imposée par le TP, vous vous apercevrez que seul le fichier *Collection.c* doit être adapté.

Pour être clair :

- Seul le fichier *Collection.c* doit être programmé
- De fait vous ne rendez que le fichier *Collection.c*
- Nous corrigerons avec nos propres versions des autres fichiers (*Collection.h*, *main.c*, ...)

Il vous est fourni les autres fichiers sauf *Voiture.c*. À la place de ce dernier fichier, il vous est fourni *Voiture.o* avec une version x86 et une version Mac.

## 3 Travail à effectuer

On impose la structure de données dans *Collection.c* : une liste doublement chaînée non circulaire de *Voiture*.

Vous devez programmer la liste entièrement, vous ne pouvez pas utiliser des bibliothèques toutes faites.

Gérez les erreurs avec des *assert* (par exemple un indice incorrect pour désigner une voiture).

Dans un commentaire en haut du fichier, mettez vos noms et prénoms. N'hésitez pas à commenter pour expliciter vos choix et vos algorithmes.