

Examen du cours I04 (AMS 2018-2019)

Christophe Labourdette

Novembre 2018

1 Indications

Cet examen a pour but de vous évaluer à la fin du cours I04, voici quelques points qui seront particulièrement importants.

- Dans la mesure du possible on essaiera d'écrire du C++ moderne utilisant par exemple des fonctionnalités du standard C++11,
- On privilégiera toujours la librairie standard,
- Les programmes doivent compiler et s'exécuter, un programme qui ne compile pas ne sera a priori pas noté.
- Les programmes peuvent et doivent être commentés.
- Les documents et la consultation sur internet est autorisée mais attention, la recopie flagrante ou le plagiat d'un programme existant sera sanctionnée, de même que les communications avec un autre candidat ou une personne extérieure.
- Les remarques et les instructions particulières seront placées dans un fichier "Readme".
- Le fichier Makefile fourni, permet de compiler question par question. Par exemple pour la deuxième question du carré : "make carre2". L'exécutable s'appelle alors carre2.exe.
Le makefile permet également de supprimer les exécutables : "make clean".
- Les fichiers sont fournis sous forme d'archive, pour l'extraire vous pouvez, une fois copiée l'archive dans votre répertoire de travail, exécuter la commande :

```
tar xvf Examen_I04_2018.tar
```

Cela créera alors un répertoire Examen_I04_2018 que vous pourrez renommer selon la consigne suivante.

- Les fichiers réponses à votre examen doivent se trouver dans un répertoire, celui-ci devra être dans votre répertoire de travail et s'appeler "Examen_I04_2018_Nom_Prenom".
(En remplaçant bien entendu Nom et Prenom par les vôtres)
- Lorsque vous aurez terminé vous devrez créer une archive (APRES AVOIR SUPPRIMER LES FICHIERS EXECUTABLES), par exemple avec la commande suivante :

```
make clean ;  
cd ; tar cpfz I04_2018_Nom_Prenom.tgz Examen_I04_2018_Nom_Prenom
```

Ensuite vous devrez copier cette archive sur la clé du surveillant et envoyer ce fichier par mail, à Christophe Labourdette.

(Christophe.Labourdette (at) cmla.ens-cachan.fr)

2 Des carrés

On considère disposer d'un carré (matrice), composé de *double* entre 0 et 1. La taille du carré est un paramètre. Dans un premier temps toute la classe sera déclarée dans une partie publique.

1. Proposez une classe Carre, , comportant, dans un fichier carre1.hpp, en plus des données :
 - un constructeur par défaut (la taille sera 1),
 - un constructeur prenant la taille du carré,On surchargera également, à l'extérieur de la classe, l'opérateur "«" pour afficher le carré.
On utilisera main1.cpp pour tester cette question.
2. A partir de cette question, on placera les données dans une partie privée.
Ajouter (on utilisera à présent un fichier carre2.hpp) :
 - un constructeur par recopie,
 - un opérateur d'assignation,
 - une fonction put permettant de changer la valeur en i,j,
 - une fonction get permettant de récupérer la valeur en i,j,
 - une fonction taille() qui renvoie la taille du carré.On n'oubliera pas de modifier l'opérateur "«" maintenant que les données sont privées. On testera à l'aide du fichier main2.cpp
3. Dans un fichier carre3.hpp on ajoutera les surcharges des opérateurs +,-,*, qui effectuent les opérations entre deux carrés élément par élément.
On testera à l'aide du fichier main3.cpp
4. Implémentez une fonction membre moyenne qui prend pour chaque point du carré la moyenne du point avec les 4 voisins les plus proches. Pour les bords on proposera deux méthodes pour remplacer les données manquantes sur les bords, "miroir", on prend la valeur du point symétrique, "tore", on prend la valeur en considérant le bord comme collé au bord opposé.
La méthode par défaut sera "miroir"
5. Dans un fichier carre5.hpp, ajouter un constructeur permettant de construire un carre de taille n comprenant une valeur par défaut dans tout le carré. Ajouter également la surcharge de l'opérateur *= , par un *double*.
On testera ensuite avec le fichier main5.cpp.
6. Dans un fichier carre6.hpp, proposez une classe CarreT, template, permettant de faire des carrés avec d'autres types que des double, en répondant avec cette classe aux questions 1,2,3 et 5.
On testera avec le fichier main6.cpp