

Travaux dirigés C++ n°4

Informatique

—IMAC 2e année—

Vers l'héritage : une application d'exemple

Le but de ce TD est de se familiariser avec l'écriture et la lecture dans les fichiers en C++ à travers le développement d'un chargeur et d'un exporteur. L'étudiant sera ensuite amené à découvrir l'intérêt de l'héritage au travers d'une application de traitement d'image.

► Exercice 1. Retrouvailles en images

Objectif de l'exercice : Concevoir une classe Image fonctionnelle Révisions : class

A faire En vous servant de vos connaissances et des précédents TD, vous devez coder la classe `ImageRGBU8` qui contient une image RGB codée en **unsigned char**. Créez les membres et fonctions essentielles pour son utilisation.

► Exercice 2. Charger une image

Objectif de l'exercice : Savoir lire dans un fichier en C++. Révisions : ifstream, getline, istream, read, <<

A faire Vous devez créer la fonction `loadPPM` prenant en paramètres un nom de fichier et une référence sur une `ImageRGBU8`. Cette méthode devra remplir l'image en utilisant le fichier PPM. Vous utiliserez les flux C++ pour lire le fichier.

On rappellera le format de fichier PPM

P6

#Commentaires

Largeur Hauteur

255

.

. Données binaires

.

► **Exercice 3. Écrire une image**

Objectif de l'exercice : Savoir écrire un fichier en C++. Révisions : ofstream, write, >>

A faire Vous devez créer la fonction `writePPM` prenant en paramètres un nom de fichier et une référence sur une `ImageRGBU8`. Cette méthode devra remplir le fichier PPM en utilisant l'image. Vous utiliserez les flux C++ pour écrire dans le fichier.

Essayez de lire un fichier PPM puis de l'écrire, vérifiez que vos images sont justes en affichant les fichiers produits (sous Gimp par exemple).

► **Exercice 4. Filtrage**

A faire Ajouter à votre classe `imageRGBU8` un ensemble de filtres. Vous pourrez par exemple faire :

- un seuillage
- dessiner un rond sur l'image
- supprimer des canaux de l'image
- et plein d'autres

Testez vos filtres en sauvegardant une image modifiée.