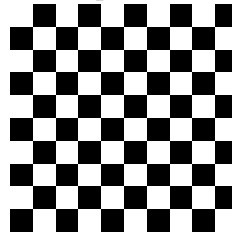


TP 9 : fichiers et images**Exercice 1.**

- (1) Télécharger l'exemple du cours `fichier-ecriture.cpp` et l'enregistrer dans votre dossier `info111`.
- (2) Exécuter ce programme. Trouver le fichier qui a été créé et l'ouvrir. Que contient-il ?
- (3) Modifier le programme et l'exécuter pour écrire un fichier nommé `essai.txt` contenant le texte « Deux puissance 10 vaut » suivi de la valeur de ce nombre.
- (4) Ouvrir ce fichier avec un éditeur de texte (par ex. `gedit`), et vérifier son contenu.

Exercice 2.

- (1) Utiliser un éditeur de texte pour écrire à la main un *fichier* contenant une image `damier.pbm` au format PBM (*Portable Bit Map*, voir http://fr.wikipedia.org/wiki/Portable_bitmap) de taille 10x10 représentant un damier :



- (2) Visualiser le résultat (vous pouvez utiliser l'application `gimp` sous Unix, ou l'application `irfanview` sous windows).

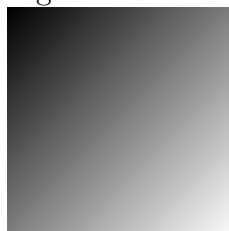
Exercice 3.

Implanter un **programme C++** qui écrit un fichier `damier-automatique.pbm` comme le précédent, mais cette fois pour un damier 100x100.

Indication : Vous pouvez vous inspirer de `fichier-ecriture.cpp`. Commencer par un programme pour un damier 10x10. Si le fichier produit ne donne pas l'image attendue, ouvrez-le avec un éditeur de texte pour mieux comprendre ce qu'il se passe et aider au débogage.

Exercice 4.

Implanter un programme qui écrit un fichier contenant une image `degrade.pgm` au format PGM (*Portable Gray Map*, voir http://fr.wikipedia.org/wiki/Portable_graymap) de taille 255 par 255, avec un dégradé de gris :



avec une image de taille 100x100 puis 1000x1000.

Répéter,

Exercice 5.

Implanter un programme qui lit un fichier contenant une image au format PGM (par exemple le fichier `image.pgm` fourni), et écrit un fichier contenant la même image en vidéo inverse (clair remplacé par sombre et réciproquement).

Indication : Implanter une fonction

```
/** Image en vidéo inverse
 * @param image1: le nom du fichier contenant l'image à lire
 * @param image2: le nom du fichier pour l'image à écrire
 */
void videoInverse(string image1, string image2);
```

Exercice ♣ 6.

Implanter un programme qui lit un fichier au format PPM (*Portable Pix Map*, voir http://fr.wikipedia.org/wiki/Portable_pixmap), et l'affiche à l'écran avec la bibliothèque MLV vue au TP 8.

Indications :

- (1) Implanter une fonction

```
/** Image en vidéo inverse
 * @param image: le nom du fichier contenant l'image à afficher
 */
void afficheImage(string image);
```

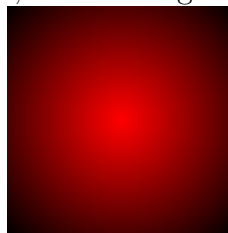
- (2) Avec MLV, on peut créer une nouvelle couleur avec la fonction suivante :

```
/** construit une nouvelle couleur
 * @param red: un entier entre 0 et 255 spécifiant le niveau de rouge
 * @param green: un entier entre 0 et 255 spécifiant le niveau de vert
 * @param blue: un entier entre 0 et 255 spécifiant le niveau de bleu
 * @param alpha: un entier entre 0 et 255 spécifiant la transparence
 */
color_t rgba(UInt8 red, UInt8 green, UInt8 blue, int alpha);
```

Par exemple, `rgba(255,0,200,128)` donnera un violet tirant sur le rouge, à moitié transparent.

Exercice ♣ 7.

- (1) Implanter un programme qui écrit un fichier contenant une image `degrade-circulaire.ppm` au format PPM de taille 255 par 255, avec un dégradé circulaire de rouge :



- (2) Répéter, avec une image de taille 100x100 puis 1000x1000.