TP 8 : tableaux multidimensionnels

Informatique Fondamentale (IF1)

Semaine du 10 Décembre 2007

1 Images noir et blanc

On rappelle qu'une image noir et blanc est représentée en mémoire par un tableau de booléens. L'affichage d'une telle image peut se faire à l'aide de la fonction suivante :

```
public static void dessine(boolean[][] image) {
    Deug.startDrawings(image.length, image[0].length);
    for(int i = 0; i < image.length; i++)
        for(int j = 0; j < image[i].length; j++) {
            if(image[i][j])
                Deug.drawPoint(i, j);
        }
}
```

(qui suppose cependant que l'image ne soit pas vide...)

1. Écrivez une fonction

```
public static boolean[][] croix(int l, int h)
```

qui retourne une image de taille $l \times h$ contenant les segments qui connectent les milieux des bords opposés.

Écrivez un programme qui affiche le contenu de cette image.

2. Écrivez une fonction

```
public static boolean[][] croixDiagonale(int l)
```

qui retourne une image de taille $l \times l$ contenant les segments connectant les angles opposés.

3. Écrivez une fonction

```
public static boolean[][] superpose(boolean[][] im1, boolean[][] im2)
```

qui retourne une nouvelle image qui contient la superposition des images `im1` et `im2`. Écrivez un programme qui affiche la superposition des deux croix dessinées aux exercices précédents.

2 Images en niveaux de gris

On rappelle qu'une image en niveaux de gris est représentée par une matrice d'entiers courts :

```
short[][]
```

contenant des valeurs comprises entre 0, représentant la couleur noire, et 255, représentant la couleur blanche.

4. Écrivez une fonction

```
public static void dessineGris(short[][] image)
```

qui dessine une image en niveaux gris.

En supposant que l'image contient des zones entières de couleur identique, comment peut-on minimiser le nombre d'appels à `setGray` ?

5. Écrivez une fonction

```
public static short[][] colourExpand(boolean[][] im)
```

qui convertit une image noir et blanc en une image en niveaux de gris identique. Testez votre fonction.

6. Écrivez une fonction

```
public static short[][] antiAlias(short[][] im)
```

qui implémente un *filtre d'anti-aliasage*. Elle retourne une nouvelle image de la même taille que l'image `im` où les " créneaux " ont été adoucis en utilisant les niveaux de gris. L'intensité d'un pixel de la nouvelle image sera obtenu en combinant la couleur du pixel correspondant de l'ancienne image avec la moyenne des couleurs des 8 pixels avoisinants, dans une proportion 3/4, 1/4.

Écrivez un programme qui affiche la version filtrée de la superposition des deux croix obtenues ci-dessus.