



RAPPELS

Dans ce TD nous allons faire quelques petites illustrations que nous sauvegarderons dans un fichier au format ppm. Le codage ppm qui permet de stocker des images graphiques. Plus précisément, vous utiliserez le format P3 qui se décrit ainsi :

```
P3 # Le P3 signifie que les couleurs sont en ASCII, et qu'elles sont en RGB.  
3 2 # Par 3 colonnes et 2 lignes :  
255 # Ayant 255 pour valeur maximum :  
255 0 0 0 255 0 0 0 255  
255 255 0 255 255 255 0 0 0
```

Le format limite le nombre de pixel par ligne, afin de vous simplifiez la vie, écrivez un pixel par ligne.

Pour les six prochains exercices, vous avez le nombre de ligne et le nombre de colonne de l'image que vous devez produire.

Exercice 1 : Écrire un programme qui produit une image aléatoire, c'est-à-dire une image dans laquelle les pixels sont choisis aléatoirement (voir figure 1a).

Exercice 2 : Écrire un programme qui produit une image qui affiche un triangle rectangle qui sépare l'image en 2 (voir figure 1b).

Exercice 3 : Écrire un programme qui produit une image qui affiche un triangle isocèle (voir figure 1c).

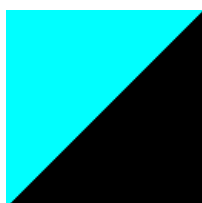
Exercice 4 : Écrire un programme qui produit une image qui affiche un drapeau (voir figure 1d).

Exercice 5 : Écrire un programme qui produit une image qui affiche un échiquier (voir figure 1e).

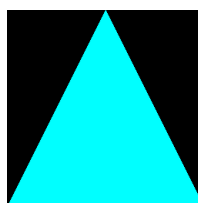
Exercice 6 : Écrire un programme qui produit une image qui du centre et le rayon, affiche un cercle de center (x, y) et de rayon r (voir figure 1e).



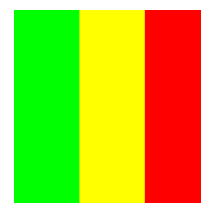
(a) Exercice 1



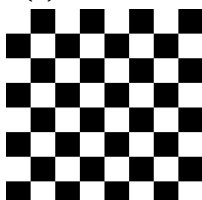
(b) Exercice 2



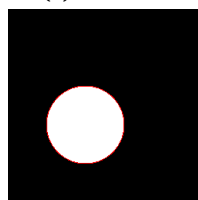
(c) Exercice 3



(d) Exercice 4



(e) Exercice 5



(f) Exercice 6

Figure 1: Illustration du résultat attendu pour les X premiers exercices.

Exercice 7 : Le jeu des bâtonnets est un jeu popularisé par l'émission de divertissement fort Boyard. C'est un jeu deux joueurs, où les joueurs retirent chacun leur tour 1, 2, ou 3 bâtonnets il ne faut pas être celui qui retirera le dernier. L'objectif de cet exercice est d'implémenter un version texte de ce jeu. Vous pouvez utiliser la fonction `scanf` pour demander une valeur à l'utilisateur.

```
| | | | | | | | | |
Joueur 1, choisir entre 1 et 3 battonets : 2
| | | | | | | |
Joueur 2, choisir entre 1 et 3 battonets : 3
| | | | |
Joueur 1, choisir entre 1 et 3 battonets : 3
| |
Joueur 2, choisir entre 1 et 2 battonets : 1
|
Joueur 1, choisir entre 1 et 1 battonets : 1
Le joueur 2 a gagné
```

Exercice 8 : Écrire une programme qui, étant données deux variables entières a et b, échange le contenu de a et b sans déclarer de nouvelles variables.

```
#include <stdio.h>

int main()
{
    int num1, num2;
    scanf("%d%d", &num1, &num2);

    printf("num1 = %d\n", num1);
    printf("num2 = %d\n", num2);

    /* le code permettant d'effectuer l'échange */

    printf("num1 après le swap = %d\n", num1);
    printf("num2 après le swap = %d\n", num2);

    return 0;
}
```

```
$ ./exec
1664
42
num1 = 1664
num2 = 42
num1 après le swap = 42
num2 après le swap = 1664
```

Exercice 9 : Écrire un programme qui vérifie si un nombre est une puissance de 2.

```
$ ./exec
Entrer un nombre : 42
Ce n'est pas une puissance de 2
$ ./exec
Entrer un nombre : 32
C'est une puissance de 2
$ ./exec
Entrer un nombre : 128
C'est une puissance de 2
```