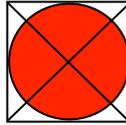


# Programmation Fonctionnelle I, Printemps 2017 – TD3

<http://deptinfo.unice.fr/~roy>

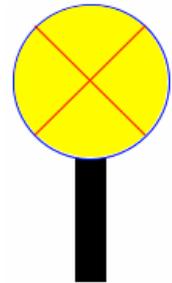
En Scheme, contrairement à Python, les **images** sont des objets à part entière, que l'on peut calculer par une fonction, ou passer en paramètre. La tortue existe aussi en Racket, nous ne l'utiliserons pas.

**Exercice 3.1** Définissez une variable `IMG` dont la valeur soit l'image d'un carré de côté 50 avec ses deux diagonales, et contenant un cercle rouge qui lui est tangent. Utilisez `underlay`... Pour le rectangle extérieur, utilisez `frame`...



**Exercice 3.2** Définissez une variable `PANNEAU` dont la valeur soit l'image d'un disque jaune à bord bleu avec deux diamètres rouges à 45°, soutenu par un piquet rectangulaire noir. Vous n'utiliserez pas d'autre variable locale que `DIAMETRE` définie ci-dessous :

```
(define PANNEAU
  (local [(define DIAMETRE (line 100 0 "red"))]
    ...))
```



**Exercice 3.3** Centre de gravité d'un triangle.

Vous savez que le centre de gravité d'un triangle est le **point de concours des médianes**, mais aussi la moyenne des trois sommets. Nous nous proposons de vérifier cela en effectuant la construction géométrique avec des images Scheme.

a) Définissez les coordonnées des sommets A,B,C du triangle :

$x_A=80, y_A=230, x_B=250, y_B=230, x_C=100, y_C=70$

b) Définissez la variable `TRIANGLE` dont la valeur est l'image du triangle ABC dont on dessine seulement les trois côtés en jaune, sur un fond noir carré de côté 300.

c) Définissez la fonction (moyenne a b) de deux nombres.

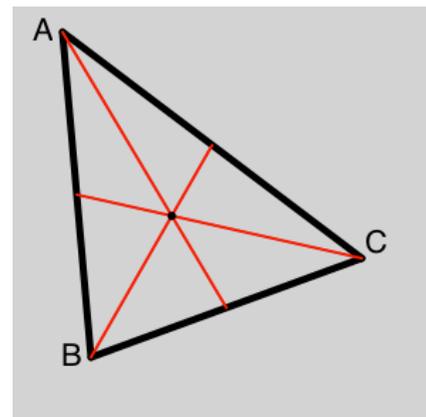
d) Soient I, J, K les milieux des côtés AB,AC,BC. Définissez les coordonnées  $x_I, y_I, x_J, y_J, x_K, y_K$  des points I, J, K.

e) Définissez la variable `TRIANGLE1` dont la valeur est l'image `TRIANGLE` à laquelle on a rajouté en rouge les trois médianes CI, BJ, AK.

f) Définissez la variable `TRIANGLE2` dont la valeur est l'image `TRIANGLE1` à laquelle on a rajouté les noms des sommets A, B, C en blanc, taille 18.

g) Définissez la variable `TRIANGLE3` dont la valeur est l'image `TRIANGLE2` à laquelle on a rajouté un petit cercle blanc de rayon 3 sur le centre de gravité G, point de concours des trois médianes.

h) Rassemblez le tout dans une fonction (`médianes xA yA xB yB xC yC`) prenant en arguments les coordonnées des sommets du triangle ABC devenu quelconque, et retournant l'image complète. Tous les objets intermédiaires seront des variables locales à la fonction !



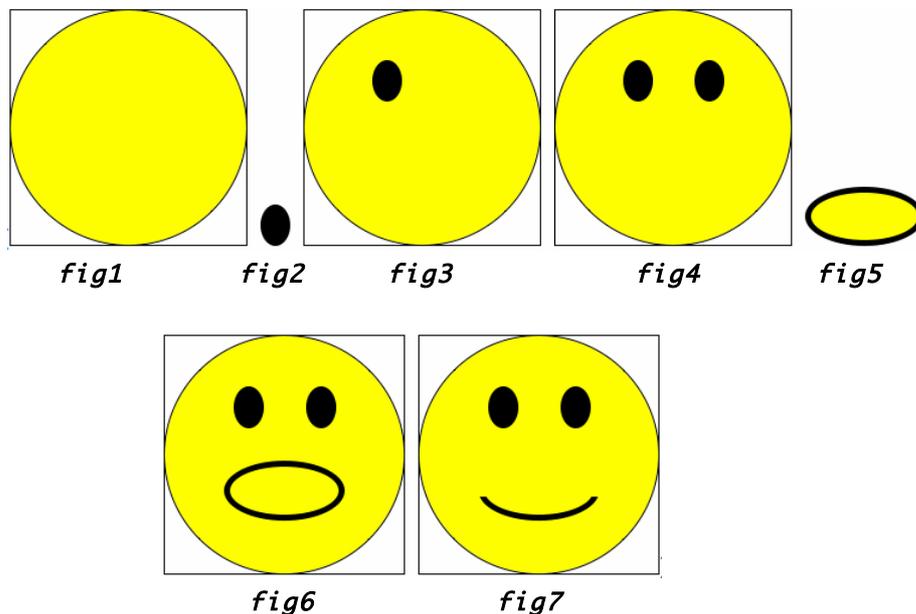
*N.B. Chez vous, exécutez ce dernier exercice sur votre machine ! Essayez de faire la même chose en traçant les hauteurs [concurrentes sur l'orthocentre H], ou bien les médiatrices [concurrentes sur le centre O du cercle circonscrit]. Il suffit de savoir exprimer une orthogonalité avec un produit scalaire... Les bissectrices sont plus difficiles.*

# Programmation Fonctionnelle I, Printemps 2017 – TP3

<http://deptinfo.unice.fr/~roy>

**Exercice 3.1** Cet exercice a pour but de construire un « smiley ». Vous allez le faire en plusieurs étapes.

- Définir une variable TETE dont la valeur soit une scène carrée de côté 200 contenant un disque jaune centré, tangent aux bords, et dont la circonférence soit un cercle noir [fig. 1].
- Définir une variable OEIL dont la valeur soit une ellipse remplie de couleur noire, de largeur 25 et de hauteur 35 [fig. 2].
- Définir une variable SMILEY1 dont la valeur soit une scène obtenue en insérant l'image OEIL dans la TETE [œil droit, fig. 3].
- Définir une variable SMILEY2 obtenue en plaçant l'œil gauche [fig. 4].
- Définir une variable BOUCHE dont la valeur est l'image d'une ellipse remplie de jaune avec une bordure noire d'épaisseur 5 [fig. 5].
- Définir une variable SMILEY3 obtenue en plaçant la bouche [fig. 6].
- En déduire enfin une variable SMILEY comme sur la figure 7, donc masquer la partie supérieure de la bouche..



**Exercice 3.2** Définir directement la variable SMILEY en une seule définition [avec un local pour les images intermédiaires] !

**Exercice 3.3** Transformez la variable SMILEY en une fonction (smiley content?) où content? est un booléen #t ou #f qui gouverne l'humeur du smiley : avec #t il est content, et avec #f il fait la tête.

**Exercice 3.4** Composez une scène intéressante et envoyez-la par mail, pour du bonus... Vous pouvez transformer un fichier image soleil.png sur le disque en une image Scheme en évaluant (define soleil (bitmap "soleil.png")). Mélangez donc de "vraies" images prises avec votre appareil photo [ou trouvées sur Internet] avec des images Scheme pour une composition originale de *réalité augmentée* : <http://www.loria.fr/~gsimon/ra/>

## OPTIONNEL

**Exercice 3.5** Reprenez l'exercice de TD 3.3 avec des nombres complexes, en programmant la fonction :

(z-médianes zA zB zC)

prenant en arguments les affixes complexes des sommets du triangle ABC.

*N.B. Vous voudrez peut-être étendre aux nombres complexes certaines fonctions sur les images. Par exemple, la fonction :*

*(add-line img x1 y1 x2 y2 color)*

*mériterait une définition complexifiée : (add-line img z1 z2 color) idem pour place-image, etc.*