

# Travail pratique de programmation concurrente

## Travail pratique - Affichage graphique

### 1 Objectif

Écrire une bibliothèque de primitives graphiques et d'interaction avec le clavier à utiliser dans un contexte multi-threadé.

Le but sera d'avoir une interface pour le jeu *Asteroids* qu'il vous faudra réaliser comme projet pour ce semestre.

Pour ce faire utilisez la bibliothèque `SDL`. Vous avez un exemple de bibliothèque `gfx` que vous trouverez sur cyberlearn (voir le travail pratique sur Julia).

### 2 Énoncé

Dans ce travail pratique il faut créer un thread qui gère l'affichage graphique et les entrées clavier de façon adéquate<sup>1</sup>. Ce thread s'occupe de créer et détruire le contexte `SDL` (voir les fonctions `gfx_create()` et `gfx_destroy()`), de l'affichage l'affichage, et de gérer le nombre de frames affichées par secondes. Pour afficher  $N$  frames par seconde, on mesure le temps,  $t$ , en secondes nécessaire pour afficher les  $N$  frames, puis si ce temps est inférieur à  $1s$ , on attend avec la fonction `usleep()` le temps  $1 - t$ . Ce thread doit également gérer les événements liés au clavier.

Simulez la logique du jeu avec une boucle `while` infinie ne faisant rien, mais permettant de mettre en place l'interaction avec le moteur graphique et les entrées clavier.

#### 2.1 Affichage

L'affichage doit être capable d'afficher trois primitives graphiques simples: un cercle (les astéroïdes), un triangle (le vaisseau), et un point (le projectile).

Pour afficher un cercle utilisez l'algorithme de Bresenham (voir l'article [wikipedia](#) correspondant).

Pour afficher un triangle, il faut afficher trois droites. Pour afficher une droite utilisez l'autre algorithme de Bresenham (voir l'article [wikipedia](#) correspondant).

Pour le point, c'est beaucoup plus simple débrouillez-vous!

---

1. Erratum un seul thread doit d'occuper de l'affichage et du clavier conformément à la doc de la `SDL` [https://wiki.libsdl.org/SDL\\_PollEvent](https://wiki.libsdl.org/SDL_PollEvent). Merci M. Glück d'avoir attiré mon attention sur cette l'erreur impardonnable qui s'était glissée dans la version de l'énoncé de la semaine passée.

### 2.1.1 Rappels de géométrie

Un cercle est décrit par la position de son centre, qui est un vecteur  $\vec{c}$ , et son rayon  $r$  qui est un scalaire. Un triangle, lui, est décrit par la position de ses sommets qui sont des vecteurs,  $\vec{t}_1$ ,  $\vec{t}_2$ , et  $\vec{t}_3$ .

N'hésitez donc pas à créer des structures adéquates pour représenter ces formes.

## 2.2 Clavier

Le clavier permet d'effectuer différentes actions:

- **esc**: le programme quitte.
- **flèche gauche**: le vaisseau tourne vers la gauche.
- **flèche droite**: le vaisseau tourne vers la droite.
- **flèche haut**: le vaisseau accélère.
- **flèche bas**: le vaisseau décélère.
- **barre d'espace**: le vaisseau tire un projectile.
- **p**: le programme se met en pause.
- **s**: le programme sauve son état.

Pour le moment ces fonctions ne sont pas implémentées. Appelez des fonctions vides en réaction à chacune de ces actions. Pour être sûr · e que vous appelez la bonne fonction affichez un message à l'écran par exemple. Le but est de mettre en place la structure de votre programme afin que la suite puisse être mise en place "facilement".