

5 juin 2023

Cappelié Lila
Teyssier Salomé
IMAC 1

Synthèse d'images 1

Projet: The IMAC Light Corridor

The Light Kittydor



1. Mode d'installation

Le dossier **/src** contient un fichier **main.cpp**, un header **draw_scene.h** qui initialise les différentes structures et fonctions, ainsi que cinq autres fichiers **.cpp** qui définissent chacun les fonctions d'un élément (setup initial du couloir, les différents obstacles, les différents bonus, les collisions de la balle, et les menus et textures). Le projet se compile avec Cmake, et est constitué dans un template compilable sur Linux et Windows.

2. Fonctionnement

Lorsqu'on lance le jeu, un premier menu s'affiche. L'utilisateur peut alors cliquer sur E pour commencer à jouer, ou sur R pour quitter. Lorsqu'il lance la partie, le départ de la balle se fait au clic gauche. Les mêmes commandes E et R sont disponibles sur les écrans de fin (gagné ou perdu).

3. Résultats obtenus

L'objectif du jeu est de garder la balle dans le couloir grâce à une raquette qui suit la souris du joueur : si la balle sort du couloir, le joueur perd une vie. Lorsque le joueur lance une partie, il possède cinq vies, qui sont affichées en bas de l'écran.

On commence à avancer dans le couloir à partir du premier tir de la balle. L'utilisateur se trouve alors face à des obstacles qui avancent vers lui, il doit faire en sorte que la balle les évite. Il y a 8 obstacles différents, dont 6 sont fixes et 2 bougent horizontalement et verticalement. 3 obstacles sont présents à la fois dans le couloir, et ils sont choisis aléatoirement.

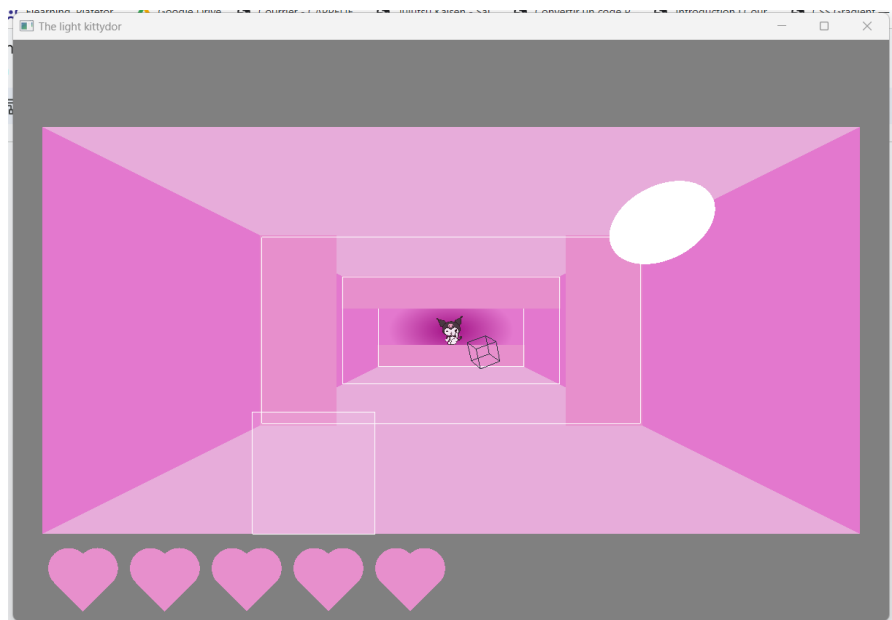
De plus, 2 bonus différents apparaissent quelques fois dans le couloir (cubes rotatifs comme dans le jeu de base), l'un permettant à l'utilisateur de gagner une vie (le joueur en a sept au maximum), et l'autre permet à la balle de coller à la raquette avant le prochain tir.

Le joueur gagne s'il parvient à passer 40 obstacles, et perd quand il arrive à 0 vie.

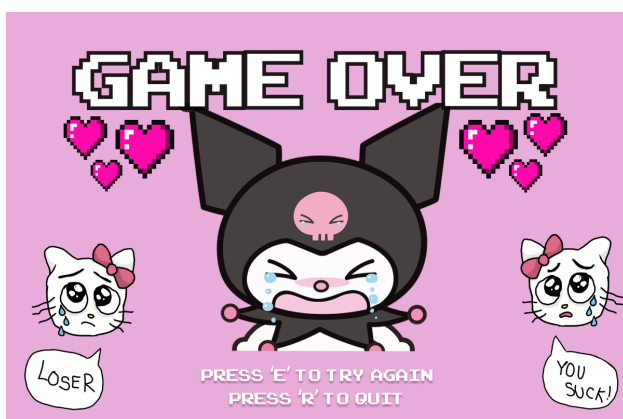
Voici quelques captures d'écran des menus et du jeu une fois lancé :



Le menu de départ



Le jeu



Les menus de fin

4. Méthode de travail

Comme nous ne travaillions pas sur le même système d'exploitation (Lila codait sur Windows et Salomé sur Mac), nous n'utilisons pas le même template et c'était parfois difficile de s'échanger nos progressions respectives. On s'est alors répartis les tâches, l'une codait plutôt les éléments et fonctions OpenGL, l'autre plutôt celles en C++.

Comme nos fichiers étaient différents pour que ça fonctionne sur nos ordinateurs respectifs, nous n'utilisons pas le GitHub (c'est la raison pour laquelle tout a été commis au même moment), et on s'échangeait nos morceaux de code au format texte pour chacune les réincorporer dans nos propres fichiers. Finalement, on a rendu le projet sur le template Windows car on savait qu'il était fonctionnel sur Linux.

5. Difficultés et Amélioration

Voici les difficultés et les améliorations que nous pourrions ajouter à notre jeu.

- Nous avons eu des problèmes avec les formats de chargement d'images pour les textures : certaines, pourtant dans le bon format, n'étaient pas lues correctement.
- Nous n'avons pas eu le temps de coder les sources de lumière, ainsi que les relations raquette-obstacles. Ici, la raquette avance même si un obstacle est juste devant elle, elle passe à travers.
- Pendant longtemps nous avons aussi eu des bugs de collisions, dus au fait que nous avons oublié une condition nécessaire au rebond de la balle. Ceci provoquait des rebonds non voulus.
- Pour la clarté du code, il aurait fallu éviter d'initialiser toutes les variables dans le main directement.
- Pour les fonctions qui chargent les textures, il aurait fallu n'en faire qu'une et passer le chemin de l'image à charger en paramètre de la fonction, au lieu d'avoir 4 fois la même fonction.
- Nous aurions pu aussi laisser quelques secondes d'écart entre le moment où la balle sort du couloir et où elle réapparaît sur la raquette, afin de bien comprendre qu'une vie vient d'être perdue. C'est la même chose pour l'affichage des écrans de fin, qui arrive instantanément après la perte de la dernière vie. Cela aurait rendu le jeu plus fluide, et simplifié la compréhension du joueur.
- Finalement, le fait que la raquette ne soit pas stoppée par les obstacles provoque parfois des problèmes de collisions : si la balle est envoyée par la raquette sur un obstacle trop proche, il arrive quelques fois qu'elle le traverse.