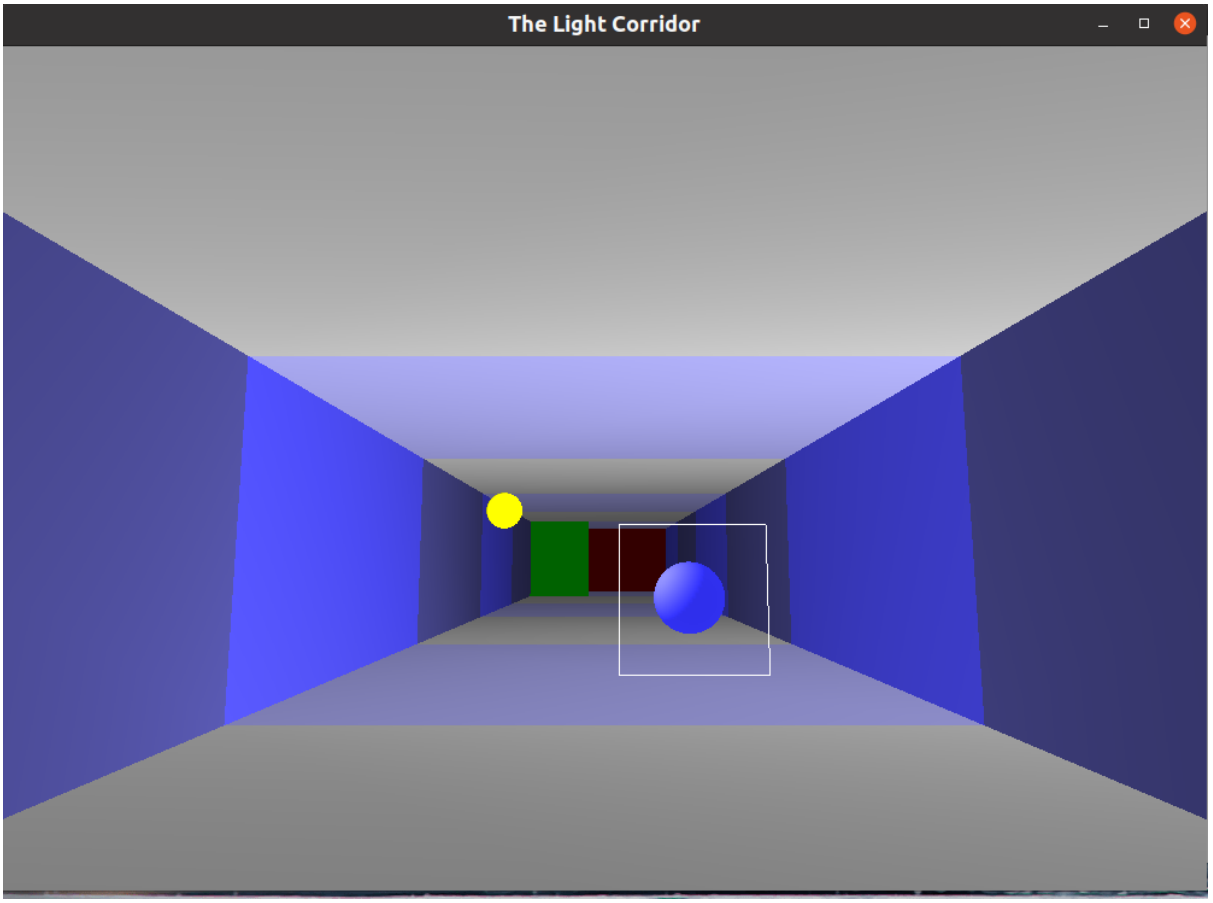
The Light Corridor

(**Nicolas ATRAX** - **Dylan DE JESUS**)



I - Dépôt github

Les fichiers C/C++ compilables se trouvent sur le dépôt github créé pour ce projet.

II - Mode d'installation

A - Compilation

La compilation de ce projet se fait au moyen de deux commandes principales en se plaçant à la racine du projet :

```
make          # Compile la version jouable/principale du projet

make test     # Compile le module de test du projet
```

B - Exécution

Une fois le projet compilé (voir section compilation), un exécutable est créé dans un dossier **bin**.
Il peut y avoir ainsi deux exécutables différents dans ce dossier :

=> tlc qui permet de lancer le jeu *The Light Corridor*
=> test qui exécute le module de test (partie développeur)

Il suffit donc de rester à la racine du projet et d'y écrire la commande suivante :

```
./bin/tlc   # Lance le jeu

./bin/test  # Lance les tests effectués
```

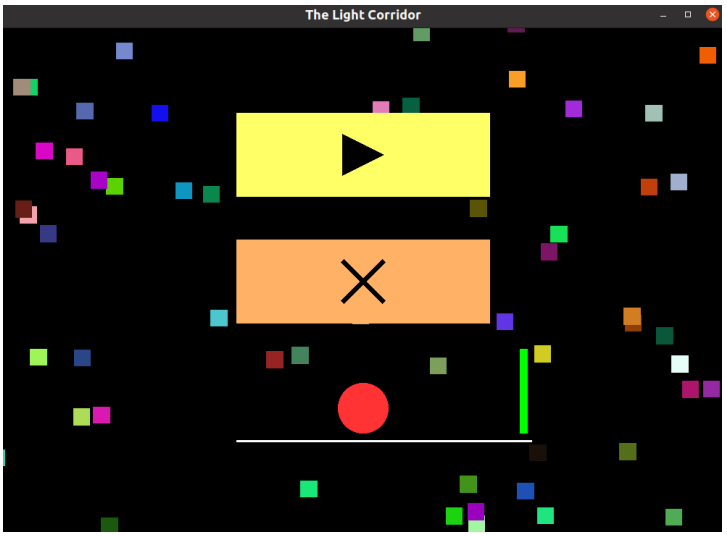
Ou en se mettant directement dans le dossier **bin** :

```
./tlc

./test
```

III - Fonctionnement/Commandes utilisateur

Le jeu se lance sur le menu de départ. Le joueur a la possibilité de lancer le jeu ou de quitter en cliquant avec sa souris sur un des deux boutons.



Une fois le jeu lancé, le joueur possède plusieurs commandes effectuant chacune une action :

=> **Clic droit** : permet de lancer la balle droit devant.
=> **Clic gauche** : permet de faire avancer la raquette droit devant sa position au moment du clic (si possible).
=> **Déplacement de la souris** : permet de bouger la raquette (sur place) en restant bien dans les limites du couloir.
=> **Appui sur la touche Echap/Q** : permet de quitter le jeu en cours.

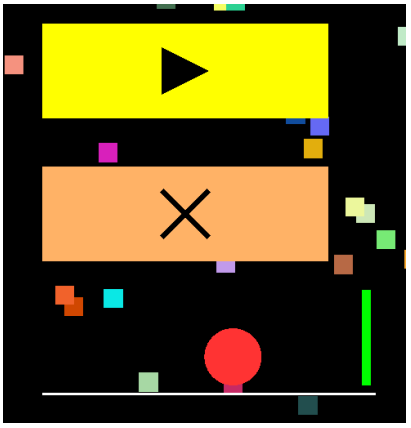
D'autres commandes sont également accessibles mais destinées à un usage développeur :

=> **Appui sur la touche B** : permet de dézoomer la caméra .
=> **Appui sur la touche N** : permet de zoomer la caméra .
=> **Appui sur les flèches directionnelles** : permet d'effectuer une rotation de la caméra.

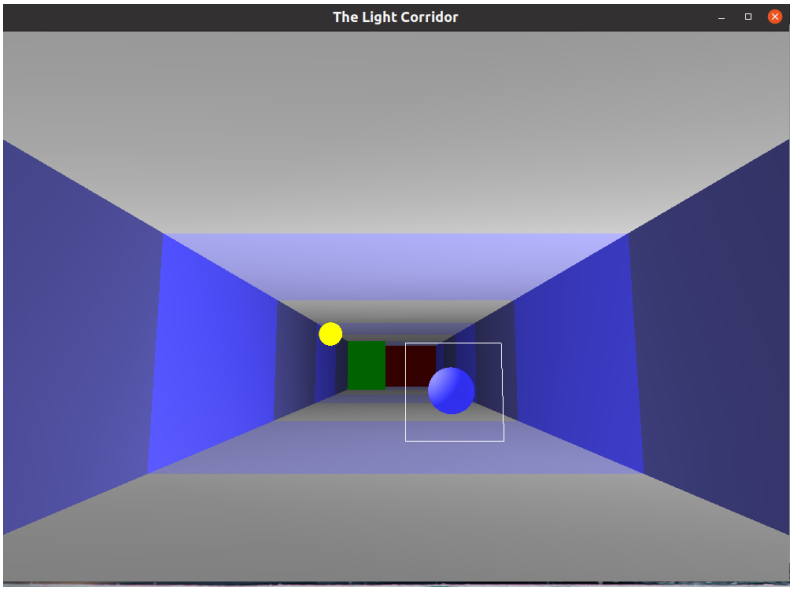
Une fois le jeu terminé, une fenêtre d'affichage permet de savoir si le joueur a perdu ou remporté la partie. Il est possible de **quitter** toujours avec la touche **escape**.

IV - Fonctionnalités implémentées/Résultats obtenus

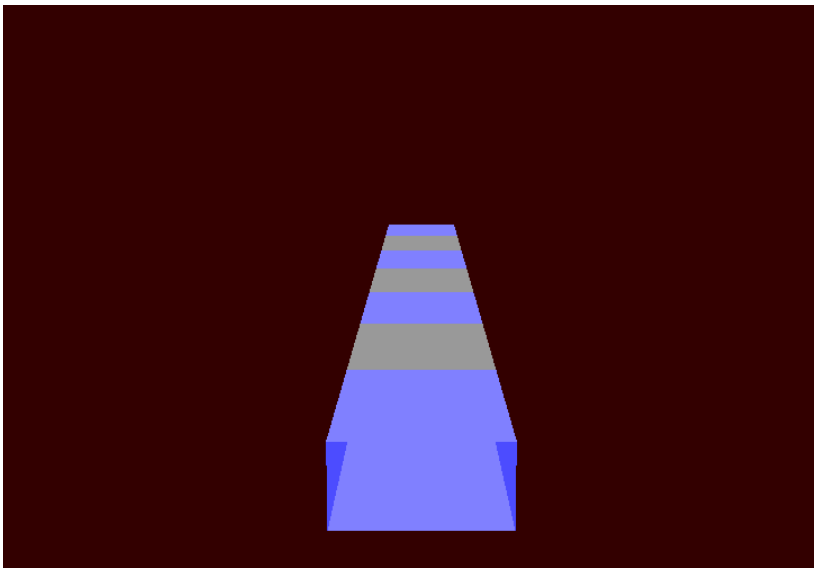
Le projet s'ouvre sur un menu en 2D qui laisse place au choix du joueur de lancer une partie ou quitter, le joueur se sert de sa souris pour choisir. Une petite animation de balle qui est stoppée par un obstacle et celle de carrés multicolores en mouvement y sont présentes.



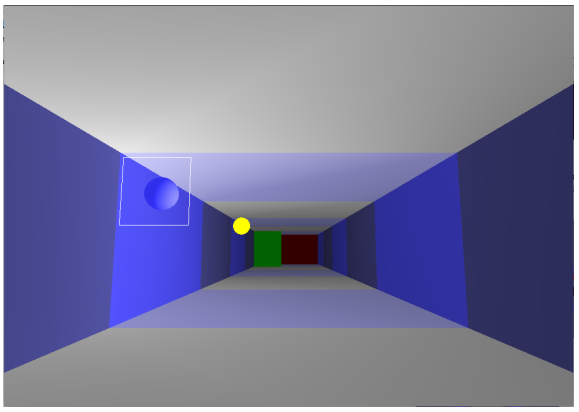
Si le joueur lance une partie, il se retrouve dans une scène 3D à l'intérieur même du couloir avec devant lui sa raquette et la balle collée.

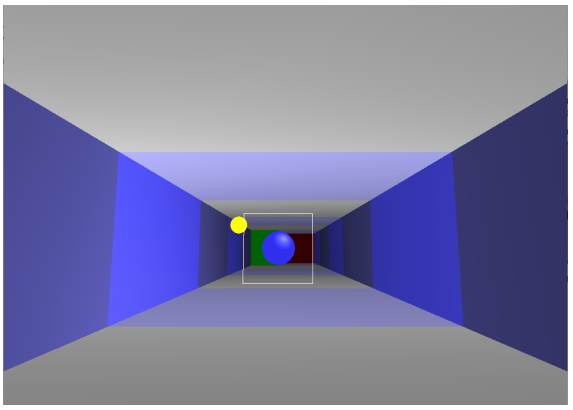


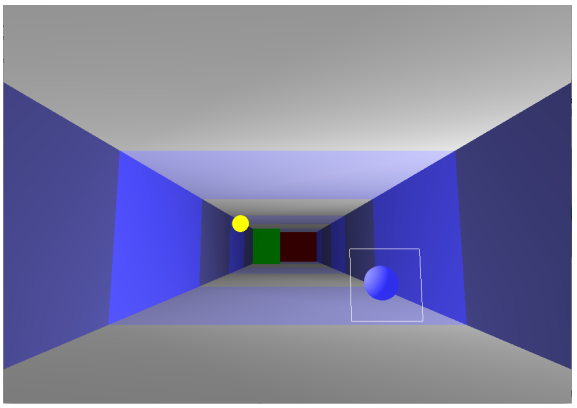
Nous avons implémenté un tunnel qui se forme progressivement jusqu'à atteindre la taille fixée aléatoirement au début du programme et qui représente la barrière virtuelle de tous les autres éléments du jeu.



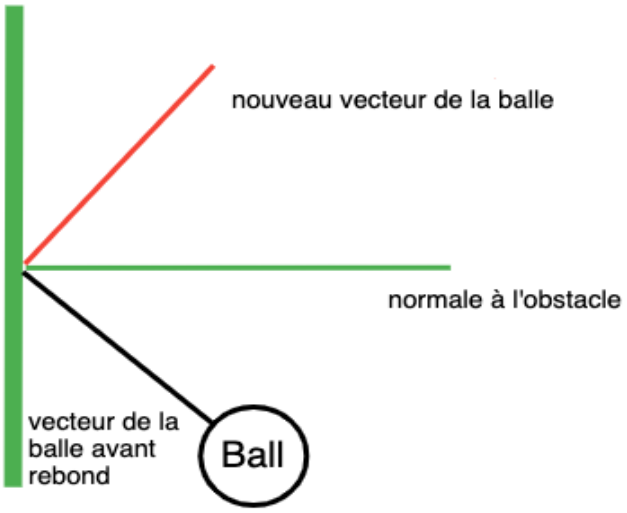
Au départ, la balle est attachée à la raquette et suit ainsi ses mouvements centrée par rapport au centre du carré qui représente la raquette du joueur.



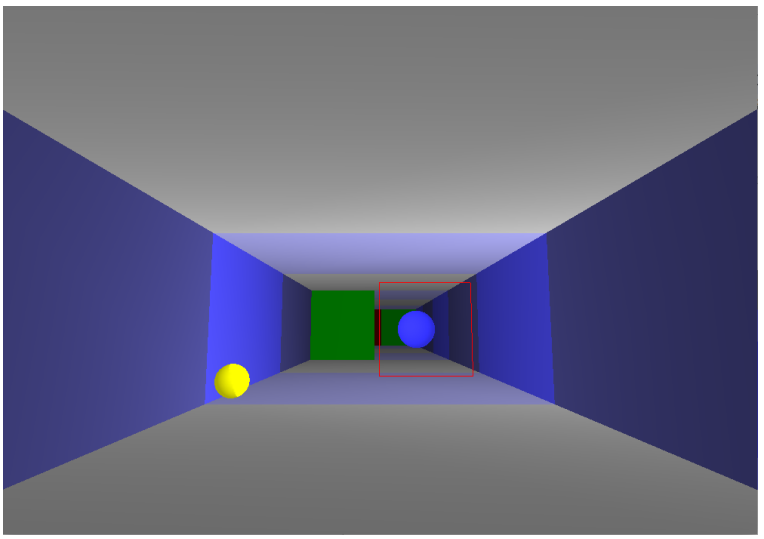




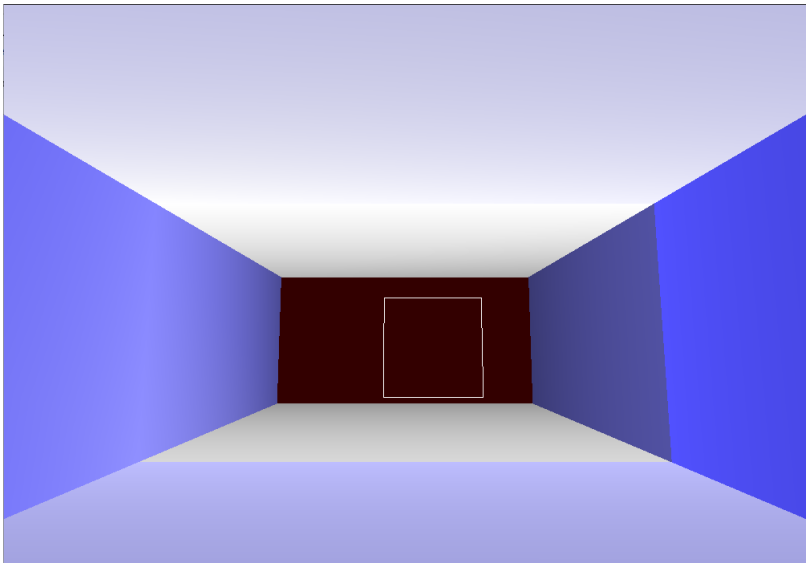
La balle rebondit sur les obstacles qu'elle rencontre (obstacles du jeu, murs du couloir, raquette) en ayant un mouvement symétrique à la normale de l'obstacle.



Nous avons également intégré des bonus représentés par des sphères jaunes qui produisent des effets positifs sur le joueur s'il les récupère en les traversant avec sa raquette. Il en existe 2, l'un permet de coller la balle au prochain contact avec la raquette (la raquette devient rouge) et l'autre ajoute une vie au joueur.

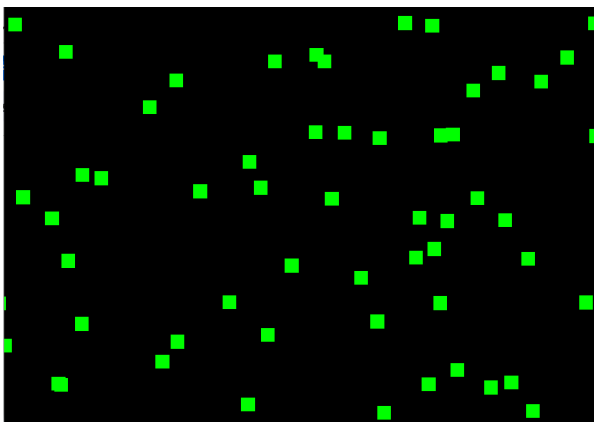


Le jeu s’arrête à deux occasions, lorsque le joueur a épuisé son capital de 5 vies (sans compter les éventuels bonus) ou lorsqu’il atteint la fin du tunnel.

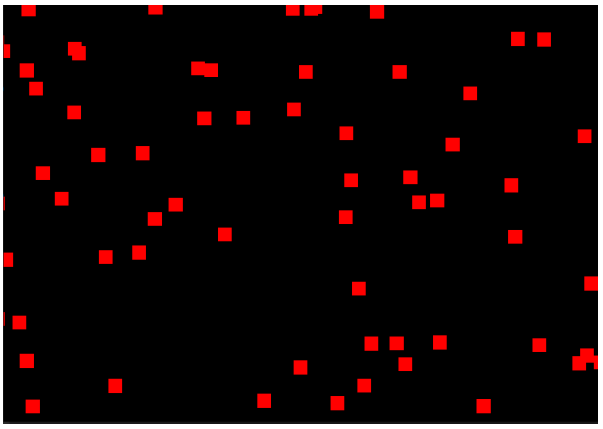


En effet, le joueur peut sortir du tunnel avec la balle devant lui qui continue son chemin, il n’existe plus de génération d’obstacles au-delà.

Enfin, le joueur est avertis de sa victoire ou sa défaite par l’affichage graphique d’une petite animation.



Victoire



Défaite

V - Détails techniques

A - Rebonds sur la raquette

Nous avons mis en place un système similaire au simple rebond de la balle sur les obstacles et murs du couloir (voir section fonctionnalités implémentées) en ajoutant un coefficient.

Nous calculons ainsi la distance entre le point de contact entre la raquette et la balle et le centre de la raquette, celle-ci détermine un coefficient amplificateur de l'angle du nouveau vecteur par rapport à la normale.

Représentation de la raquette :

```
c ----- d
|       |
|       |
a ----- b
```

Calcul du point central de la raquette :

```
double center_racket_x = racket_points.d.x - ((racket_points.d.x -
racket_points.a.x) / 2);
double center_racket_y = racket_points.d.y - ((racket_points.d.y -
racket_points.a.y) / 2);
```

Calcul de la distance point de contact -> centre de la raquette :

```
double distance_center_x = ball.x - center_racket_x;
double distance_center_y = ball.y - center_racket_y;
```

Calcul des nouvelles coordonnées de translation :

```
ball_trans_x += ball_speed * distance_center_x;
ball_trans_y += ball_speed * distance_center_y;
```

Symétrie de la translation :

```
ball_trans_x *= -1;
ball_trans_y *= -1;
ball_trans_z *= -1;
```

B - Structure des éléments cycliques

Nous avons implémenté une structure principale complexe lors de la réalisation de ce projet.

La génération cyclique d'éléments (obstacles, bonus) se base sur une liste chaînée qui se remplit progressivement au fur et à mesure que la balle avance dans le jeu et dans l'ordre de leur apparition dans le tunnel, les éléments les plus proches de la raquette seront donc en premières positions.
On retire tous les éléments de la liste dont la position en **z** a été dépassée par celle de la raquette(obstacles et bonus).

VI - Méthode de travail

Nous avons eu une méthode de travail organisée et progressive, nous nous sommes répartis des heures de travail disjointes afin de ne pas empiéter sur le code de chacun.
Ainsi, à chaque fin de session pour un membre du binôme s'accompagnait un résumé de ce qui a été fait et une discussion sur la progression du projet.

VII - Difficultés rencontrées

Lors de la réalisation de ce projet, nous avons rencontré deux difficultés majeures : le **temps** et le **fonctionnement matériel**.

En effet, il nous a été très difficile de prendre du temps au cours du semestre pour travailler sur ce projet et avions planifié de consacrer la plupart de notre temps à celui-ci une fois nos examens terminés, de plus un des membres du binôme (**Dylan DE JESUS**) n'a pas pu programmer sur son ordinateur personnel et a ainsi dû venir spécialement à l'Université pour pouvoir travailler dessus.

VIII - Améliorations possibles

L'ajout d'une texture n'ayant pas été implémenté, celui-ci est donc la prochaine amélioration envisageable.
Parmi les autres améliorations possibles sur ce projet, nous aurions également aimé rendre les formes des différents composants plus **complexes**, comme mettre plus de **volume** dans la profondeur des **obstacles**, de la **raquette**, mais également

travailler plus en profondeur les animations de fin de jeu en y ajoutant du **texte** et/ou **images** afin de rendre ces affichages plus agréables et compréhensibles pour le joueur.