

Rapport de Projet IGSD 2024

Raphael LEONARDI, Baptiste PRAS



Fonctionnalités :

- Pylônes reliés entre eux via des câbles
- Éoliennes fonctionnelles reliées aux pylônes et à la ville
- Monter, descendre, avancer et reculer dans la carte
- Rotation sur soi-même vers la droite, gauche, bas, haut
- Système de collisions avec la texture
- Déplacement automatique en hauteur lorsque l'on heurte les montagnes
- Shader avec lignes de niveau et coloriage de la vallée en vert
- Ajustement au clavier de la hauteur de la vallée et de la taille des lignes
- Système de vue « FPS » (la vue suit la courbure de la texture) et système de vue global (on se déplace où l'on veut dans l'espace, sauf hors des murs de la texture en x+, x-, y+, y- et z-)
- Activation/Désactivation du shader, de l'affichage des coordonnées, du repère et des collisions au clavier
- Musique de fond, musique easter egg lorsque qu'on tourne sur soi-même

Structure du code :

- Code structuré en 3 fichiers : main.pde, pylone.pde et eolienne.pde
- Classes et PShape pour les objets pylone et eolienne
- Déplacements gérés dans main.pde
- Fichiers vertexShader.glsl et fragmentShader.glsl pour le shader

Fonctionnement des shaders :

- Fichiers vertexShader.glsl et fragmentShader.glsl
- vertexShader.glsl modifie la couleur du vertex actuel de la texture
- fragmentShader.glsl récupère cette donnée sur le vertex, récupère la texture via un sampler2D et applique selon la hauteur l'une des 3 possibilités :
 - 1) En hauteur, on applique un filtre gris multiplié par la texture pour griser les montagnes
 - 2) Si l'on est sur une ligne de niveau, on met un trait noir
 - 3) Sinon, on applique un filtre vert multiplié par la texture pour verdir la vallée
- On obtient la hauteur actuelle du fragment dans fragmentShader.glsl via vertPos.z et on calcule le z avec la profondeur du fragment modulo 2, permettant de créer deux niveaux
- On passe au shader deux valeurs uniformes depuis le main, « ajustement » et « taille », qui déterminent respectivement à quelle hauteur on place la différence entre vallée et montagnes (vert et gris) et l'épaisseur des lignes de niveau. On peut modifier ces valeurs dans le main depuis le clavier une fois le programme lancé.
- On applique le shader uniquement s'il est demandé, c'est-à-dire si l'utilisateur après avoir lancé le programme n'a pas demandé à le désactiver au clavier

Difficultés :

- Beaucoup de temps passé sur les shaders, on n'arrivait pas à multiplier les vertex par la texture, on avait donc soit une texture toute verte/grise, ou transparente, mais jamais un filtre appliqué à la texture de base
- Difficultés à obtenir les points en Z de la shape pour positionner les pylônes et éoliennes en fonction d'un X et Y donnés