Universidad de Chile
FCFM - DCC
CC3501

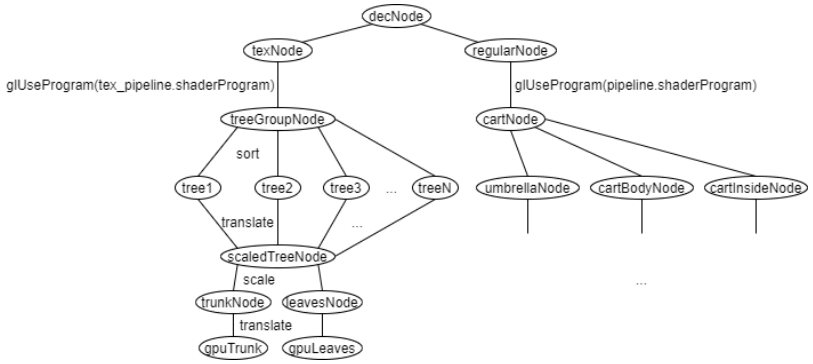
Reporte Tarea 1a

Andrés Abarca Muñoz
10 de mayo de 2021

Solución Propuesta

Se modifica el funcionamiento de grafos de escena y algunos shaders, estos módulos modificados se guardan en la carpeta custom_modules.

Se crean distintos grafos de escena que modelen diferentes partes del escenario, como el fondo y las decoraciones. El grafo usado para las decoraciones es modificado para ser capaz de dibujar cada nodo con un pipeline distinto. Para evitar cambiar de pipeline muy seguido, en el grafo utilizado se crean dos nodos hijos de la raíz, uno para albergar todos los nodos hijos que deben ser dibujados con un pipeline, y el otro para los nodos que deban ser dibujados con un pipeline distinto.



Ejemplo: Segmento del grafo de escena de decoración

Se definen clases para modelar el comportamiento del jugador, la tienda, los humanos y los zombies, y a cada objeto creado se le asocia un nodo de grafo de escena que será su modelo. Se crean Z zombies y H humanos cada T segundos en la parte superior o inferior de la pantalla, y junto con el jugador se guardan en una lista de entidades. Los zombies y humanos tienen movimientos aleatorios, y los humanos pueden o no aparecer infectados. Tanto los objetos de tipo Humano como el de tipo Player, se convierten en zombies si colisionan con uno, y se infectan si colisionan con un Humano infectado. Además al estar infectados, cada T segundos se genera un numero al azar entre 0 y 100, y si es menor a P * 100, se transforman en zombies creando un objeto de tipo Zombie colocándolo en la lista de entidades en su lugar. La lista que contiene los modelos de la entidades se actualiza para eliminar aquellos nodos que estén fuera de la pantalla, luego transforma en hijo de un grafo de escena, y este se dibuja.

Cuando el jugador se encuentra infectado se aplican transformaciones de shearing que simulan efecto de mareo, además, se dibuja una espiral sobre su cabeza cuyos vértices se modifican y transforman en CPU.

Se crean dos shaders personalizados, uno asigna las coordenadas de textura segun la posición del objeto para dar un efecto distinto del usual, y el otro recibe un parámetro uniform que modifica el color de la figura según el tiempo que lleva el objeto infectado, para ser usado como efecto de los lentes detectores.

Se crean nuevos objetos con texturas de "Game Over" y "You win!" para indicar que el juego ha acabado, dependiendo de si el jugador se ha convertido en zombie, o ha avanzado suficiente distancia para que se comienze a dibujar el grafo de escena de la tienda.

Instrucciones de Ejecución

- Librerias adicionales a OpenGL utilizadas: sys, os.path, math, random, numpy, PIL
- Método de Ejecución: python survival.py Z H T P
- Modo de uso y/o teclas de control: W, A, S, D, Espacio, Escape

Z y H son la cantidad de zombies y humanos a liberar en escena cada T segundos, y P es la probabilidad de que un humano infectado se transforme en zombie cada T segundos. Los parámetros ingresados Z, H y T son de tipo int, y el parámetro P es un número de tipo float entre 0 y 1.

El jugador se mueve con las teclas "W", "A", "S" y "D", para las direcciones arriba, izquierda, abajo y derecha respectivamente, se pueden mantener presionadas para movimiento constante. La barra espaciadora activa y desactiva los lentes detectores de la infección al ser presionada. El juego puede ser cerrado con la tecla "Escape".

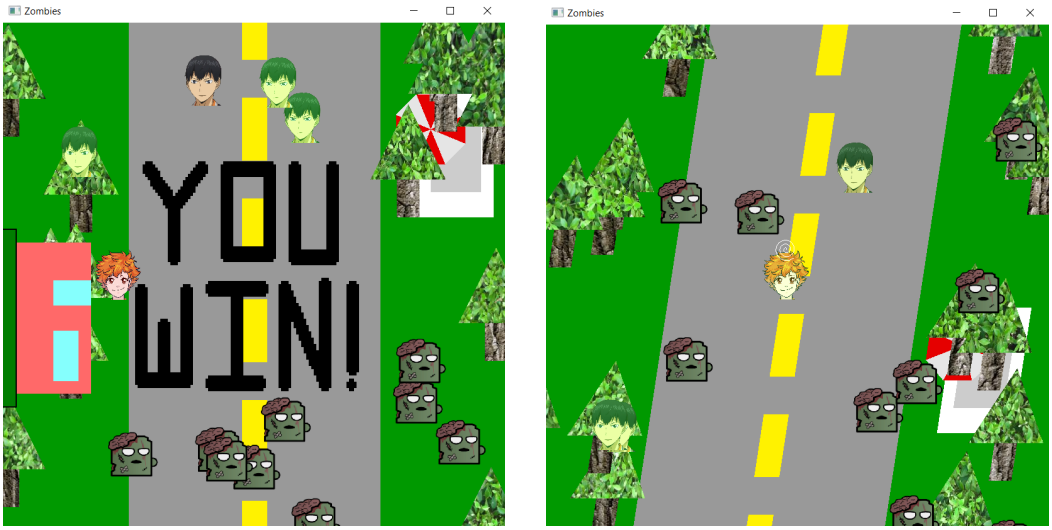
Universidad de Chile
FCFM - DCC
CC3501

Reporte Tarea 1a

Andrés Abarca Muñoz
10 de mayo de 2021

Resultados

Se crean todos los grafos de escena que modelan el fondo y las decoraciones, al avanzar la distancia suficiente se comienza a dibujar la tienda que dará la victoria al colisionar con el jugador. Si el jugador está en la mitad de la pantalla, al avanzar, el resto de grafos de escena se mueven y el jugador se mantiene estático, dando las senación de movimiento. Con los lentes activados, los humanos infectados se observan con un tono más verde, y si el jugador se infecta, se distorsiona la pantalla y una espiral giratoria se dibuja en la cabeza del jugador.



Ejemplos: Victoria del jugador al colisionar con la tienda (izquierda), y el jugador infectado, con la pantalla distorsionada (derecha)

Autoevaluación

Criterio-Puntaje	0	1	2	3
OpenGL				x
Shaders			x	
Modelos geométricos			x	
Transformaciones				x
Texturas				x
Modelación jerárquica				x
Curvas	x			
Funcionalidades mecánicas o lógica de juego				x
Entradas o Control de usuario			x	
Visualización de estado del programa			x	