

# Tarea 1: OpenGL en 2D: Space War

CC3501 – Modelación y Computación Gráfica para Ingenieros

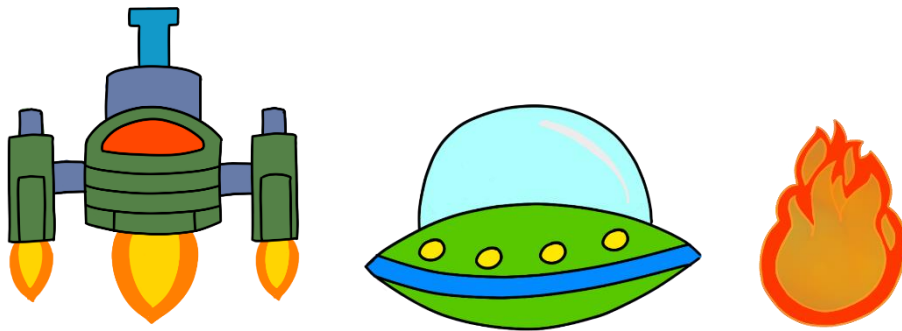
09-05-2020

Cristóbal Saldías Salgado

En esta tarea se desarrollará un juego llamado “Space War” en el cuál se maneja una nave y el propósito es eliminar a todos los enemigos para ganar el juego.

1. Partimos llamando al módulo desde la consola con *“python space-war.py N”* con N el número de enemigos.

Definimos funciones auxiliares que generen las GPUs con el uso de texturas y nodos, se usan 4 sprites para hacer la animación de la nave aliada, 1 solo Sprite para las naves enemigas y 1 sprite para los disparos enemigos.



2. Luego de inicializar glfw, se lee el número de enemigos gracias a *“int(sys.argv[1])”*, se crea el fondo, los disparos hechos por el usuario, la nave y las N naves enemigas con sus respectivos disparos. También se crean las GPUs para mostrar cuando ganemos o perdamos y una pequeña interfaz que muestra las vidas que nos quedan.

3. Al momento de entrar en el Loop para actualizar la ventana, el fondo se mueve para dar la sensación de ir avanzando, se anima la nave cada 0.5 seg y se cambia su posición con las teclas WASD (presionando o manteniendo), impidiendo que esta se salga de la pantalla. Al pulsar espacio se hace True una variable que nos ayuda a crear una bala cada vez que se cumpla este valor y que se guarde en nodos para mostrarla en la ventana.

Las N naves van apareciendo desde la parte superior de la pantalla cada 3 segundos y son invulnerables hasta llegar a la altura 0.7, aquí se mueven sinusoidalmente en los ejes X e Y mientras realizan disparos que bajan y se mueven en X. Para hacer que las naves vuelvan a disparar, cuando la bala haya pasado la parte inferior de la pantalla o nos impacte, se traslada hacia la altura de la nave para repetir el proceso.



(N =5 Enemigos disparando // Enemigos y yo disparando)

Después hacemos que nuestros disparos se dibujen subiendo mientras no se salgan de la pantalla o impacte a un enemigo, en ese último caso es cuando ambas posiciones en X e Y son muy cercanas, entonces ambos dejan de existir y por ende ya no se dibujan.

Por último, actualizamos el contador de vidas cuando nos impacten y lo mostramos en pantalla. Al perder las 3 vidas aparece la pantalla de Game Over y si matamos a todos los enemigos muestra la pantalla de victoria.

