



Kind of... Asteroids?

18.05.2017

PARDO GARCÍA, FEDERICO

Tecnología de la programación

Grupo 1.3

Teoría: JUAN ANTONIO SÁNCHEZ LAGUNA

Prácticas: GRACIA SANCHEZ CARPENA

Convocatoria: JUNIO 2017

****/ Indice**

**/ Indice	1
/** Descripción	1
Objetivo	1
Funcionamiento general	1
Diario de desarrollo (Resumido)	2
/** Manual de usuario	2
/** Organización del proyecto	2
Main.c	2
Pantalla.h	2
Bala.h	3
Listabalas.h	3
Enemigo.h	3
Listaenemigos.h	3
Personaje.h	3
Colisiones.h	3
/** Estructuras de datos	4
Lista de Balas:	4
Array de enemigos:	5
/** Conclusiones	5

/** Descripción

Objetivo

El objetivo del juego es sencillo. Mantenerse con vida el máximo tiempo posible mientras te amenazan hordas de enemigos. ¿Cómo hacerles frente? Tienes una pistola que te permite defenderte. Sus disparos son eficaces. Con un solo proyectil, eliminas a cualquier enemigo. Pero sólo puedes disparar una bala en un mismo segundo. No se obtiene puntuación por eliminar enemigos, pero lo necesitarás si quieres sobrevivir.

Funcionamiento general

El juego emite aleatoriamente una serie de enemigos. Estos rebotan por la pantalla sin tener ninguna finalización más que el impacto de uno de los proyectiles del jugador. Por su parte, el jugador será eliminado en cuanto uno de los enemigos lo toque. Es la única condición de finalización del juego. Las balas están limitadas a una en un mismo segundo para aumentar la dificultad del juego.

Diario de desarrollo (Resumido)

El código del juego ha tenido que ser reescrito unas 4 veces en total, debido a muchos problemas con CodeBlocks y sobretodo con el compilador. La última de éstas fue la buena porque decidí limpiar mi ordenador al completo y con ello el compilador y el programa. La versión entregada no es la más avanzada en el desarrollo. Hay algunas funciones extra que, al no estar finalizadas, no están en ésta versión y son reservadas para el desarrollo personal. Entender que el juego ha sido simplificado para cumplir mínimamente con los criterios que se me piden para él. Mi versión personal es más eficiente y con más funciones (de nuevo, en desarrollo).

*/

/** Manual de usuario

Al aparecer la primera pantalla, la tecla para avanzar al juego es "esc". Una vez dentro, las teclas W, A, S y D se encargaran de mover al personaje por la pantalla, mientras que el ratón indica la dirección a la que disparar y el click izquierdo lanza una bala en esa dirección.

*/

/** Organización del proyecto

Los archivos que componen el programa son los siguientes.

- **Main.c**
 - Este archivo no depende de ningún otro, ya que es el encargado de llevar todas las funciones y hacer que todos los archivos puedan trabajar entre sí.
- **Pantalla.h**
 - Contiene parte de la librería de SDL adaptada por la Universidad para la realización de este proyecto, simplificando el nombre de las funciones, así como haciéndolas más accesibles.
- **Bala.h**
 - El encargado de administrar las balas de manera individual. Solamente se declara en Listabalas.h y en Colisiones.h, ya que en ambos casos se necesita acceder a funciones individuales de la bala.
- **Listabalas.h**
 - Únicamente llama a Balas.h, ya que se encarga de administrar listas de balas en base a funciones de éste fichero.
- **Enemigo.h**
 - Fichero encargado de administrar a los enemigos de forma individual. Declarado, al igual que Balas.h en Colisiones.h y en Listaenemigos.h, por los mismos motivos.
- **Listaenemigos.h**
 - Llama a Enemigo.h, ya que se encarga de administrar listas de enemigos en base a funciones de éste fichero.
- **Personaje.h**
 - Encargado de administrar a personajes que controle el jugador. En éste caso concreto, solo controla 1.
- **Colisiones.h**
 - El fichero más importante del juego. Controla las colisiones entre balas, enemigos y el personaje. Llama a todas las funciones anteriores salvo Pantalla.h y Main.c

*/

/** Estructuras de datos

En este proyecto apenas se han utilizado 2 estructuras de datos muy sencillas. Una lista, utilizada para las balas, y un array para los enemigos (dado que se nos obligaba a utilizar ambos). En cuanto a los restantes Tipos de Datos Abstractos, no hay nada relevante que comentar, por lo que me centraré en las dos estructuras.

I. Lista de Balas:

La lista de balas está compuesta por una cabecera y un campo siguiente NULO. EN caso de añadir una bala a la lista, ésta se introduce por el principio. La función más interesante es Lista_Balalimpia, cuya única función es eliminar las balas que se encuentren fuera de la pantalla.

```
void Lista_Balalimpia (Listabalas l)
{
    while ( l->sig != NULL )
    {
        if( (0>Bala_getx(l->sig->b) || Pantalla_Anchura()<Bala_getx(l->sig->b))
        || (0>Bala_gety(l->sig->b) || Pantalla_Altura()<Bala_gety(l->sig->b)) )
        {
            Listabalas eliminado = l->sig;
            l->sig = eliminado->sig;
            Bala_Libera(eliminado->b);
            free(eliminado);
        }
        else
        {
            l = l->sig;
        }
    }
}
```

II. Array de enemigos:

El array de enemigos se compone de los campos en los que se introducen los enemigos y 2 enteros. El primero dedicado para llevar la cuenta de cuántos enemigos hay activos en el array y el otro para indicar el máximo de enemigos posibles.

*/

/** Conclusiones

El desarrollo de este juego está basado en Asteroids. La razón de esto no es que sea uno de los juegos recomendados para llevar a cabo como proyecto, sino porque fue el primer videojuego al que jugué en mi vida con apenas 2 años en el ordenador de mi padre. Y por ello pensé: "¿Qué mejor juego para hacer como primer programa de verdad?" Así que de ahí salió la idea de desarrollar esto. No se parece en nada al original e incluso me faltan funciones por incluir para poder hacerle sombra, pero no tengo tiempo físico para poder desarrollar nada más, así que he de dejarlo aquí. Puede que en verano lo finalice como proyecto personal.

*/