

PROYECTO #1A. MANUAL TÉCNICO.

**VIDEOJUEGO SHOOTER
TANQUES.**

PROFESOR. Víctor Garro Abarca.

ESTUDIANTES Edgar André Araya Vargas.
Diego Moscoa Dobles.

CARNÉS 2020142856.
2020362145.

GRUPO 02.

CICLO LECTIVO Semestre II-2021

INDICE.

Principales Estructuras.....	3
1- struct BalaTanque	3
2- struct TanqueEnemigo	3
Modulo de Pantalla.	3
3- void pantalla()	3
Modulos de Dibujar.	3
4- void dibujarMapa()	3
5- void dibujarAzul().....	4
6- void dibujarPuntuacion()	4
7- void dibujarTextoInicial()	4
8- void dibujarEnemigo()	4
9- void dibujarBalasTanque().....	4
Módulos de Sonido.	5
10- void reproducirMusicaJuego()	5
11- void reproducirDisparoTanque().....	5
12- void reproducirLadrilloRoto().....	5
13- void reproducirExplosionTanque()	5
Módulos para Mover.....	6
14- void moverAzul()	6
15- void moverEnemigo()	6
16- void moverBalaTanque(int movimiento).....	6
Módulos de Estructuras.....	7
17- void iniciartanquesEnemigos().....	7
18- void desactivartanqueEnemigo(int posicion)	7
19- void iniciarBalasTanque()	7
20- void desactivarBalaTanque(int posicion).....	7

PRINCIPALES ESTRUCTURAS.

1- **struct** BalaTanque

Estructura encargada del evento-cola de las balas que dispara el jugador. Esta estructura tiene de atributos un valor entero x, un valor entero y, un valor entero que la da la orientación en la que esta siendo disparada la bala, y un valor booleano que define si la bala esta activa o no, llamado activado.

2- **struct** TanqueEnemigo

Estructura encargada de los tanques enemigos que se presentan en el juego. Estos tienen de atributos un valor entero x que define su posición x en la pantalla, un valor entero y que define su posición y en pantalla y un valor entero de dirección que nos dice que dirección lleva cada tanque enemigo.

MODULO DE PANTALLA.

3- **void** pantalla()

Modulo encargado de limpiar la pantalla y actualizarla con todos los elementos agregados. Para esto tomamos le asignamos el buffer del display, se dibujar, llamamos al `flip_display()` y al `clear_to_color(transparente)`.

MODULOS DE DIBUJAR.

4- **void** dibujarMapa()

Modulo encargado de dibujar el mapa en la pantalla. Para esto le damos dos valores enteros llamados fila y columna. Utilizamos estos dos valores para recorrer el mapa/matriz anteriormente

creado con dos ciclos for, uno dentro de otro, donde se revisa cada casilla de la matriz. Si la casilla de la matriz tiene una “X” se dibuja un bitmap de una imagen de un ladrillo con allegro.

5- void dibujarAzul()

Modulo encargado de dibujar el tanque principal que será controlado por usuario. Esta toma un buffer que se creara en main del tanque azul y en este dibujara con allegro la sección específica de un Sprite dependiendo de su dirección. Es decir, si va hacia la derecha pintara la sección de la imagen donde el tanque mira hacia la derecha. Se llama pantalla() para refrescar la pantalla.

6- void dibujarPuntuacion()

Modulo encargado de dibujar la puntuación que lleva el jugador en la pantalla. Para esto se crea un char llamado puntajeCadena que toma el valor dentro del atributo “puntaje”. Así nuestro valor entero pasa a ser una cadena y podemos imprimir el texto en la posición que queramos en nuestra pantalla con al_draw_text(). Se llama la función pantalla().

7- void dibujarTextoInicial()

Modulo encargado de dibujar al inicio del juego un texto que le dira al jugador que hacer. Este solo imprime el texto en pantalla por unos segundos y luego se quita. Se llama la función pantalla().

8- void dibujarEnemigo()

Modulo encargado de dibujar los enemigos del juego. Este revisa un arreglo de tanques enemigos y si la posición es diferente a nulo, dibuja el enemigo. Dibuja igual que al tanque principal para esto toma un buffer de tanque enemigo y dibuja un bitmap encima del tanque enemigo. Utiliza un Sprite y la dirección de cada tanque del arreglo para decidir como dibujarlo.

9- void dibujarBalasTanque()

Modulo encargado de dibujar las balas dentro del arreglo balasTanque. Para dibujar las balas toma la posición x y y que tiene cada bala específicamente y la dibuja ahí mismo.

MÓDULOS DE SONIDO.

10- `void reproducirMusicaJuego()`

Modulo que crea una instancia de sonido que toma un sonido especifico ingresado desde el main y utiliza las funciones de allegro para reproducirlo en el juego. Reproduce la música que sonara durante todo el juego.

11- `void reproducirDisparoTanque()`

Modulo que crea una instancia de sonido que toma un sonido especifico ingresado desde el main y utiliza las funciones de allegro para reproducirlo en el juego. Se reproduce cada vez que el tanque dispara.

12- `void reproducirLadrilloRoto()`

Modulo que crea una instancia de sonido que toma un sonido especifico ingresado desde el main y utiliza las funciones de allegro para reproducirlo en el juego. Se reproduce cada vez que un ladrillo se rompe.

13- `void reproducirExplosionTanque()`

Modulo que crea una instancia de sonido que toma un sonido especifico ingresado desde el main y utiliza las funciones de allegro para reproducirlo en el juego. Se reproduce cada vez que un tanque explota. Es decir, cuando el usuario destruye un tanque o cuando nuestro tanque es destruido.

MÓDULOS PARA MOVER.

14- `void moverAzul()`

Modulo encargado de mover el personaje dependiendo de que dirección tenga. En el main el jugador presionara las teclas direccionales para dar una dirección, usando el atributo `direccionAzul`. Mediante una serie de condicionales esta función se encarga de mover al tanque principal. Esto aumentando los atributos que definen su posición x y su posición y. No se permitirá que el jugador traspase paredes o salga de la zona de juego.

15- `void moverEnemigo()`

Modulo encargado de mover al enemigo dependiendo de la dirección que tenga. A diferencia del personaje principal los enemigos tienen un algoritmo donde si encuentran con pared se moverá en otra dirección. También se encarga de finalizar el juego si uno de los tanques enemigos llegara a alcanzar al tanque principal del jugador.

16- `void moverBalaTanque(int movimiento)`

Modulo encargado de mover las balas disparadas por el tanque principal manejado por el usuario. Esta función se llama cada vez que el jugador presione la tecla espacio. Es el código mas grande del juego. Se revisa el arreglo que contiene las balas disparadas por el jugador y si es diferente a nulo se va a una serie de condicionales que definen en qué dirección se moverá la bala. Una vez con la dirección se define que si choca con un enemigo lo destruya y este desaparezca, que si encuentra una pared se destruya y la bala desaparezca y que si encuentra el borde del juego la bala desaparezca. Aquí se aumenta el puntaje cuando choca con enemigos o bloques. Si no choca con nada, se mantiene en su dirección y se aumenta para que continúe en movimiento.

MÓDULOS DE ESTRUCTURAS.

17- `void iniciartanquesEnemigos()`

Modulo encargado de iniciar las estructuras de los tanques enemigos dentro de un array de tanques enemigos. Para esto se genera un valor random de “x” y un valor random de “y” que sirvan para la posición donde serán generados los tanques en el mapa de juego. Así siempre será diferente. Se recorren las 10 posiciones del array de tanques enemigos y se les asigna su posición “x” su posición “y” y la dirección que tendrán.

18- `void desactivartanqueEnemigo(int posicion)`

Modulo encargado de desactivar un tanque enemigo una vez que fue destruido. Para esto tomamos el parámetro tomado de posición, tomamos esa posición del array tanquesEnemigos y lo declaramos como Nulo. Aquí aumenta la estadística de tanques destruidos y también se reduce el numero de cantidad enemigos en pantalla.

19- `void iniciarBalasTanque()`

Modulo encargado de iniciar las estructuras de la lógica de balas. Para esto se recorre el array de balasTanque y se les asigna un valor de null. Mas tarde este valor de null será remplazado cuando se dibuje la bala.

20- `void desactivarBalaTanque(int posicion)`

Modulo encargado de desactivar las balas del tanque una vez que choque con un objetivo o algo de pantalla. Para esto tomamos la posición en la estructura de balasTanque de la bala especifica y se le asigna como NULL.

21- `void masEnemigos()`

Modulo encargado de generar mas enemigos. Para esto replicamos el código de `iniciarTanquesEnemigos()`, donde creamos posiciones random que tomarán estas estructuras y serán agregadas al array de tanques enemigos. Esto solo pasara si la cantidad de enemigos es menor a 5, así el jugador puede jugar infinitamente contra los tanques enemigos y mejorar su puntuación. Nos encargamos de definir que los enemigos no aparezcan encima o cerca del jugador principal.

22- `void dispararTanque(int direccion)`

Modulo encargado de generar balas de nuestro tanque. Es decir, en las posiciones inicializadas de un array, se genera una bala dependiendo de la dirección en el parámetro. Esa dirección viene de la dirección que tenia el tanque en ese momento. Así la bala puede crearse en la posición correcta para luego ser dibujada y movida por otros módulos.