TEXTO DESCRIPTIVO

Imágenes: ../imagenes/..

Audios: ../audio/..

Proyecto: ../Ping-Pong.cbp

INTRODUCCION

El juego es un clon del Ping-Pong hecho en C++ bajo la biblioteca Allegro.h

Está vinculado a bibliotecas personalizadas las cuales almacenan clases, funciones, prototipos, bitmaps y samples de audio.

No tiene utilización de bibliotecas de terceros. Sino la de allegro 4.2.1

- Dimensiones: 1000x500 bits.
- Píxel color: 24.

INICIALIZACIONES

El mecanismo del juego se centra en un TIMER el cual esta instalado (install_timer) e inicializado bajo el comando BPS_TO_TIMER (ticks por segundo) con parámetro 70. Esto controla la velocidad del juego. El tiempo de ejecución es de 70 ticks por segundo. El timer esta controlado bajo la función incremento el cual lo incrementa en 1.

A su vez tiene instalado el teclado (install_keyboard) y el sonido (install_sound).

Cuenta con 7 tipos de sonidos .WAV distintos y 11 bitmaps. 1 buffer central, printeado en la screen con dimensiones (1000x500 bits) y 10 bitmaps personalizados cargados desde imágenes .BMP.

MAIN

Los BITMAPS centrales son los de la Portada, Jugador1 gana, y Jugador2 gana. Estos son printeados en el screen con dimensiones (1000x500 bits) una vez que se llama al disparador correspondiente para luego tener la opción de cerrar o seguir el juego. Los BITMAPS restantes son característicos del bucle del juego (while), el Buffer está basado en color verde oscuro (pantalla principal del juego, generada mediante código hexadecimal), las paletas de ping-pong son bitmaps cargados y personalizados mediante aplicaciones de escritorio.

La portada se muestra junto a la inicialización de un sample de audio de introducción. Esta contiene los datos del juego junto a las reglas (resumidas). El main cuenta distintas funciones/prototipos tomadas y desarrolladas de la clase vinculada juego.h y juego.cpp, estas son Inicializaciones(), Loop_del_juego() y Destructores().

El loop principal del juego contiene los movimientos del Player1 y del Player2. A su vez la carga de paletas, carga de niveles, carga de pelota, carga de tablero y mesa.

El juego tiene un tiempo de ejecución/interlocución de CPU de 20 (rest(20)), dándole al programa una forma de juego fluida.

HEADERS Y CPPs

El juego cuenta con 4 clases distintas (Jugadores, Pelotas, Paletas y Funcionalidades) y 2 headers (Dibujar y Juego). Estás están configuradas bajo Getters y Setters y se declaran arduamente en el prototipo Inicializaciones(), mayormente con las coordenadas, tamaños y posiciones de las paletas y la pelota, como también la velocidad en que se mueve la pelota (indistinta de la velocidad del timer).

La biblioteca Pelotas.h declara la clase Pelota y facilita la ejecución de Pelotas.CPP, la cual contiene las funciones de Direccion, Reset, y Movimiento de ella. Dentro de estas se llama a los BITMAPS Jugador1gana y Jugador2gana, los cuales actúan como pantalla de Triunfo/GameOver, como así también a los samples de audio (golpe, saque, punto, golpe pared). Contiene las funciones que le dan función al juego, son sobre todo cálculos de coordenadas y posiciones, dirección de la pelota y posición de las paletas. Las demás bibliotecas/clases contienen Getters, Setters, funcionalidades del juego, destructores de BITMAPS, destructores memoria dinámica, y estructuras.

La clase Funcionalidades.h contiene las funcionalidades /ventana, portada, nuevo, resultado, tiempo/ - /disparos, propulsión/ - y los contadores /nivel, contador/. Esta clase facilita la vista y análisis del programa.

INSTRUCCIONES

El juego cuanta con un TIMER el cual facilita su jugabilidad.

El ganador es el que llega primero a 10.

Hay cambios de velocidad (aumentos) cada 4 saques.

Cada jugador cuenta con 5 disparos especiales (aumento de 1 acumulado en velocidad / Velocidad = Velocidad + 1 (loop) /)

En el nivel 3, los puntos se reinician a 5, estén o no en 0.

Una vez finalizado el juego o llegado a los 10 puntos. Los jugadores tienen la opción de jugar nuevamente (ENTER) o salir de este (ESC).

Cada BITMAP que es disparado durante la partida tiene sus respectivas instrucciones.