

## **Trabajo Práctico: Fase 3 - Juego Completo**

### **Objetivo**

Desarrollar la funcionalidad completa del juego de acción de naves espaciales.

### **Personajes principales**

Cada nave principal comienza el juego con 3 vidas, teniendo 100 unidades de energía por vida. Se deben visualizar las barras de energía de todos los personajes principales.

### **Colisiones**

No existen colisiones entre las naves principales.

En el caso que una nave principal colisione con una nave enemiga, ambas son destruidas.

### **Proyectiles**

Las naves principales dispondrán de la acción de disparo ilimitado.

Los disparos seguirán una trayectoria horizontal .

### **Enemigos**

Implementar cierto comportamiento inteligente para las naves enemigas, en donde busquen a las naves principales para atacarlos.

### **Puntos**

Cada jugador acumulará puntos al destruir naves enemigas, dichos puntos deben visualizarse en pantalla durante la partida.

- Cada nave enemiga de tipo 1 otorga 500 puntos.
- Cada nave enemiga de tipo 2 otorga 1000 puntos.

### **Energía**

- Cada nave enemiga de tipo 1 inicia con una energía total de 100 unidades.
- Cada nave enemiga de tipo 2 inicia con una energía total de 200 unidades.
- Cada disparo recibido por una nave enemiga le resta 50 unidades de energía.
- Cada disparo recibido por una nave principal le resta 15 unidades si provino de nave enemiga tipo 1, y le resta 25 unidades si provino de una nave enemiga tipo 2.

### **Pantalla de finalización de niveles**

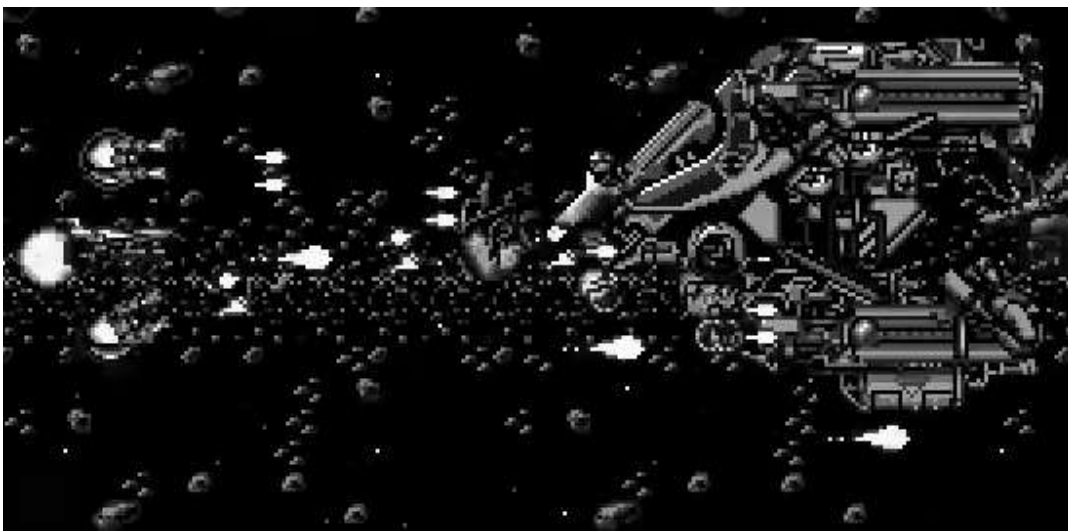
Al finalizar el primer nivel se mostrará un resultado parcial de puntajes de los jugadores hasta ese momentos.

Al finalizar el juego se mostrará el resultado parcial y final de los puntajes obtenidos por los jugadores.

### **Enemigo Final**

Se debe implementar un enemigo final (similar al de la imagen) el cual tiene una energía de 1000 unidades.

La destrucción del enemigo final otorga 2000 puntos.



### **Música y Sonidos**

Se debe implementar música de fondo para el transcurso del juego, su activación/desactivación deberá realizarse mediante la pulsación de una tecla en el medio de la partida.

Se deben implementar sonidos para los disparos, los impactos de los disparos y las destrucciones de naves.

### **Modo Test**

Mediante una tecla se podrá activar o desactivar el modo test. El mismo consiste en que los personajes no mueren una vez removida toda su energía. El mismo será utilizado para propósitos de prueba.

## Configuración

El archivo de configuración será un archivo de texto plano en formato json, el cual contendrá los parámetros de configuración inicial de la aplicación. En caso de no encontrar el archivo, se debe leer un archivo de configuración por defecto.

**Ejemplo de configuración:**

```
{
  "configuration": {
    "log": {
      "level": "ERROR"
    },
    "game": {
      "enemies": [
        {
          "type": "class_1",
          "quantity": "5"
        },
        {
          "type": "class_2",
          "quantity": "10"
        }
      ],
      "stages": [
        {
          "fondo1": "/home/file21.bmp",
          "fondo2": "/home/file22.bmp",
          "fondo3": "/home/file23.bmp"
        },
        {
          "fondo1": "/home/file31.bmp",
          "fondo2": "/home/file32.bmp",
          "fondo3": "/home/file33.bmp"
        }
      ],
      "users": [
        {
          "username": "juan",
          "password": "123abc"
        },
        {
          "username": "federico",
          "password": "qwerty"
        }
      ]
    }
  }
}
```

Pueden agregarse la cantidad de parámetros que se consideren necesarios, como así también darle la estructura que consideren necesario al archivo json.

## Restricciones

- La implementación deberá estar hecha en C/C++.
- Para la lectura y escritura de archivos JSON debe utilizarse una biblioteca.
- Para el manejo de gráficos se deberá usar la biblioteca SDL.
- Todo el código debe ser desarrollado íntegramente por cada grupo. No se permite la reutilización de código de cuatrimestres anteriores o de otras materias. Ante cualquier duda se deberá consultar con los docentes. La reutilización de código sin consulta previa será condición suficiente para la desaprobarción de la materia.