Desarrollo de Juegos para Dispositivos Móviles (CC64)

**Examen Final 2020-01**

Profesor: Luis Valdivia Humareda

Secciones: Todas

Duración: 110 minutos

*No escribir en los siguientes recuadros*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| *Pregunta 1*  *4* | *Pregunta 2*  *4* | *Pregunta 3*  *4* | *Pregunta 4*  *4* | *Pregunta 5*  *4* | *=* | *Nota*  *20* |

**En los siguientes recuadros indicar su número de cuadernillo y máquina:**

|  |  |
| --- | --- |
| **Número de**  **cuadernillo:** | **Número de**  **máquina:** |

**Instrucciones**:

* Se puede utilizar material de consulta e Internet, la solución debe ser realizada con Sublime Text con XAMPP o Visual Studio Code.
* Código que no compila no será considerado en su totalidad.
* Usar los assets brindados
* Todos deben ser Prefabs

# Pregunta 1 (2 puntos)

Crear la siguiente estructura:

* El juego debe medir 800x600
* Debe adaptarse a todas las pantallas
* Utilizar como fondo la bg\_layer4, en todos los states
* Preload
  + Cargar todas las imágenes
  + Las medidas para los spritesheet son las siguientes:
    - dude => 32x48
* Menu
  + Poner un texto con el título “Examen Final”, tamaño 20 px, color negro
  + Poner un texto Jugar
  + Jugar lleva al menú de selección
* Selección
  + Muestra dos opciones como texto: Normal y Parao sin polo, ambos llevan al Game
* Game
* GameOver

# Pregunta 2 (2 puntos)

* En selección si se elige Normal, el usuario jugará con 4 vidas, que se representan en una caja de texto.
* En cambio, Parao y sin polo, solo contará con una vida

# Pregunta 3 (2 puntos)

* Usar el spritesheet de dude como personaje principal
  + Debe poder moverse dentro del escenario usando las flechas del teclado
  + Debe poder saltar con la flecha de arriba
  + Usado el tile dentro de la carpeta floor, generar un grupo de plataformas que sirvan como piso.

# Pregunta 4 (12 puntos)

El juego consiste en conseguir la mayor cantidad de puntos sobreviviendo ola tras ola.

Si todo sale del escenario debe desaparecer

**Olas (4 puntos)**

Primera Ola: 4 enemigos de tipo red cada 8 segundos

Segunda Ola: 8 enemigos entre red y yellow cada 6 segundos

Tercera Ola: 16 enemigos entre red, yellow y fly cada 4 segundos

Cuarta Ola: 32 enemigos entre red, yellow, fly y Brown cada 3 segundos

Quinta Ola o superior: 64 enemigos entre red, yellow, flyw,Brown y cream cada 2 segundos

**Disparos (2 puntos)**

El player puede disparar diversos tipos de balas:

* A => bullets/ Brown.png, mata Browns
* S=> bullets/cream.png, mata cream
* D=> bullets/flying.png, mata fly
* F=> bullets/red.png, mata red
* G=> bullets/yellow.png, mata yellow
* Las balas de cada tipo salen del héroe y se dirigen hacia los enemigos
* Cada bala destruye un tipo de enemigo, si los tipos coinciden los dos desaparecen.
* Si chocan tipos distintos solo la bala desaparece.

**Puntaje (2 puntos)**

* Brown5 puntos
* Red 10 puntos
* Yellow 15 puntos
* Cream 20 puntos
* Fly 50 puntos
* Representar el puntaje en un caja de texto
* Guardar el puntaje en localstorage por nivel

**Enemigos (4 puntos)**

* Red y Yellow salen de izquierda a derecha o viceversa a la altura del piso
* Cream sale de la parte superior del escenario, su posición en X debe ser aleatoria entre todo el escenario, se mueve de arriba hacia abajo
* Fly salen de izquierda a derecha o viceversa a la mitad del escenario
* Brown sale de la parte inferior del escenario, su posición en X debe ser aleatoria entre todo el escenario, se mueve de abajo hacia arriba
* Si los enemigos chocan con el personaje deben desaparecer y pierde una vida
* Si la vida es 0 mandar al GameOver

# Pregunta 5 (2 puntos)

* Mostrar como título Game Over y el puntaje obtenido
* Luego de 3 segundos mandar al Menu

# Rúbrica de Calificación

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Categoría | Excelente | Aceptable | Deficiente | Puntaje |
| 1 | Implementa los 5 states del juego con sus características | Implementa un par de states del juego | No usa states dentro de su proyecto |  |
|  | (2 puntos) | (1 puntos) | (0 puntos) |  |
| 2 | Implementa funcionalidad de elegir dificultad | No implementa este state |  |  |
|  | (2 puntos) | (0 punto) |  |  |
| 3 | Implementa todas las funcionalidades básicas del player | Implementa a medias todas las funcionalidades del player | No implementa las funcionalidades del player |  |
|  | (2 puntos) | (1 puntos) | (0 puntos) |  |
| 4 | Implementa a generación de olas correctamente | Implementa solo un par de características de todas las olas | No implementa las olas |  |
|  | (4 puntos) | (2 puntos) | (0 puntos) |  |
| 5 | Implementa lógica de disparos | Implementa un par de disparos | No implementa los disparos |  |
|  | (2 puntos) | (1 puntos) |  |  |
| 6 | Implementa lógica de enemigos | Implementa solo un par de comportamientos de enemigos | No implementa ningún comportamiento |  |
|  | (4 puntos) | (2puntos) |  |  |
| 7 | Implementa Gameover | No implementa gameover |  |  |
|  | (2 puntos) | (0 puntos) |  |  |