

SPACE INVADERS

Beade, Di Toro, Lopresto Ripa, Professi

GUÍAS RÁPIDA Y EXTENDIDA PARA LA CREACIÓN DE NIVELES

Guía extendida para la creación de niveles	1
Guía rápida para la creación de niveles	5
Level Tester	7

Guía extendida para la creación de niveles

Introducción

Durante la ejecución del juego, los niveles creados en la carpeta se cargan a medida que se avanza en la partida, permitiendo así diferentes dificultades y niveles muy variados. Los mismos pueden ser modificados o creados por el usuario siguiendo esta guía.

Assets

Los asset son aquellos objetos, tales como las naves, las barreras, que aparecerán durante todo el desarrollo del juego, interactuando con los demás en cada nivel.

Por ejemplo, las siguientes imágenes corresponden cada una a un asset distinto



Sin embargo, si se tienen dos objetos con las mismas características, por ejemplo, dos aliens iguales en un mismo nivel, estos corresponden a un único asset.

En definitiva, **el asset define un “tipo” de objeto que aparecerá durante el nivel.** Todos los objetos deberán pertenecer a un asset y la dependencia del mismo se realiza mediante un **número de identificación ID.**

Este número está definido en cada archivo de cada asset y permite de manera rápida y sencilla definir a qué asset corresponde un objeto que se encuentra en determinado punto de la pantalla. Esto definirá con cuántas vidas inicia, cuál es su sprite, entre otras características.

Dichas características se enumeran a continuación:

```
1 id 11           ;Identificacion del asset
2 vel 10          ;Velocidad
3 ancho 54        ;Ancho
4 alto 39         ;Alto
5 initLives 2     ;Vidas iniciales
6 shootProb 3     ;Probabilidad de disparo
7 maxBullets 7    ;Cantidad maxima de balas
8 balaID 31       ;ID correspondiente a la bala que dispara
9 ;Sprites correspondientes a cada estado de animacion
10 sprite1 game/spritesAllegro/alien11.png
11 sprite2 game/spritesAllegro/alien12.png
12 sprite3 NULL
13 sprite4 NULL
14 score 100      ;Score asignado al usuario cuando es asesinado
15 aliado 0       ;Diplomacia
```

1. ID: Mediante un número identifica al asset para que pueda ser referenciado en los archivos de cada nivel.
2. Vel: Dependiendo de cual sea la función del asset, la velocidad define cual es el salto en X o en Y cada vez que se llama en el hilo correspondiente, su signo determina para qué lado se mueve (negativo es hacia arriba). Nota: En los aliens, debido a que se deben mover al mismo ritmo, la velocidad no se tiene en cuenta y se usa otro valor que será nombrado más adelante.
3. Ancho y Alto: Representa el “hitbox” del asset correspondiente.

SPACEINVADERS - Guía para la creación de niveles

4. InitLives: Representa con cuántas vidas va a ser inicializado el objeto cuando "nazca".
5. shootProb: Representa la probabilidad de disparar una bala cada vez que se ejecuta el hilo correspondiente a disparar la bala. Nota: En la nave nodriza representa la probabilidad de aparecer.
6. maxBullets: Máxima cantidad de balas disparadas por la nave que pueden aparecer en simultáneo.
7. balalD: Indica que tipo de bala dispara el asset. Nota: Si su valor es 0, no puede disparar ninguna bala.
8. Sprites: Estos sprites cambian dependiendo del estado de animación de cada objeto vivo. Dependiendo de cuál es su función en el juego pueden cambiar dependiendo de las vidas, o dependiendo de su movimiento. Por ejemplo, cada vez que un alien se mueve, el sprite que se muestra en pantalla va variando entre el sprite1 y el sprite2. Sin embargo, cuando una barrera pierde una vida, el sprite a mostrar cambia de 1 a 2, de 2 a 3 y de 3 a 4 dependiendo de la cantidad de vidas perdidas.
9. Score: Determina cuántos puntos se le asignará al usuario cuando ese asset sea destruido.
10. Aliado: Si es 1 se trata de un asset aliado, si es 0 se trata de un asset enemigo.

Carga de los Assets:

Los assets son cargados al principio de la ejecución del juego, todos los archivos correspondientes a la plataforma (rpi_ en el caso de la Raspberry Pi o Inx_ en linux) son leídos, el nombre del archivo de cada asset no influye. Es importante que no existan dos asset con el mismo ID.

LevelSettings:

El levelSettings es una gran variable que incluye constantes y variables que determinan el funcionamiento del juego en un determinado nivel. Dichas variables determinan por ejemplo cuáles son las coordenadas que limitan el tamaño del nivel, las velocidades de ejecución que determinan el movimiento de balas, naves, usuario. Cuántas veces puede aparecer una nave nodriza en un nivel, entre otras indicadas a continuación.

```
1 anchoUsr      60 ;Ancho de la nave del usuario
2 desplazamientoUsr 20 ;Desplazamiento del usuario en cada toque de tecla
3 desplazamientoX 20 ;Desplazamiento de las naves de los aliens en cada salto
4 desplazamientoY 20 ;Desplazamiento de las naves en Y cuando toquen el borde
5 velAliens     100 ;Velocidad del thread que mueve los aliens
6 velBalas      5   ;Velocidad del thread que mueve las balas
7 margenX       1   ;Margen de pixeles ignorados en el eje X
8 margenY       100 ;Margen de pixeles ignorados en el eje Y
9 xMax          900 ;Posicion maxima en X
10 xMin         0   ;Posicion minima en Y
11 yMax         760 ;Posicion maxima en Y
12 yMin         0   ;Posicion minima en Y
13 usrLives     0   ;Contador de vidas del usuario
14
15 mothershipYpos 50 ;Posicion en Y de la mothership
16 mothershipAsset 14 ;Asset correspondiente a la mothership
17 maxMShipXLevel 3  ;Cantidad maxima de motherships por nivel
18 velMothership 20  ;Velocidad del thread que mueve y crea la mothership
19
```

Dichos campos de level settings se pueden modificar en cada archivo de nivel, dicho proceso se explicará más adelante.

SPACEINVADERS - Guía para la creación de niveles

Niveles:

Los niveles están separados por archivos los cuales se encuentran en la carpeta game/levels.

- *Nivel 0:*

Durante la inicialización del juego, es decir, al comienzo de la ejecución, el programa busca el nivel número 0. Dicho nivel es obligatorio para el funcionamiento del juego, en el caso en el que no se encuentre se mostrará un mensaje en consola indicando el error.

El nivel 0 debe contener únicamente la inicialización de todos los campos de levelSettings. Dicho nivel es necesario debido a que define constantes que son necesarias para el backEnd como por ejemplo las coordenadas límites del juego, y la velocidad de ejecución de algunos hilos.

- *Nivel N:*

Estos niveles son cargados únicamente cuando se seleccionan. Determinan qué asset y en qué posición va a aparecer.

La indentación en la lectura de estos archivos no es importante. Se pueden comentar líneas utilizando un ; seguido de un comentario, así como también dejar líneas en blanco. Además, tampoco es importante la separación entre el parámetro y su valor, es decir, puede haber una tabulación, un espacio, o una combinación de tabulaciones y espacios.

Modificadores

Los modificadores permiten cambiar una constante/variable al inicio del nivel que fue previamente definida. Los valores modificables son cualquier campo de levelSettings (a excepción de las constantes que definen las coordenadas límite xMax,xMin,yMax,yMin) y cualquier campo de cualquier asset.

Modificación de levelSettings:

Para modificar un campo de level settings basta con poner en cualquier línea del nivel que esté fuera de la inicialización de cualquier asset, el nombre de la variable y su nuevo valor.

```
12 maxMShipXLevel      3
13 velMothership       20
14 ;-----ALIENS FILA 1-----
15
16 ALIEN
17     type  1
18     posX  150
19     posY  100
20 END
```

Por ejemplo, en este caso se estaría modificando maxMShipXLevel cambiando su valor anterior por 3, y velMothership cambiando su valor anterior por 20

También, se pueden establecer modificaciones relativas, es decir, se puede cambiar el valor de una variable teniendo en cuenta su valor anterior. Las posibilidades se muestran a continuación:

SPACEINVADERS - Guía para la creación de niveles

```
148 velBalas      +=-1    ;;Se resta 1 a velBalas
149
150 velMothership *=11    ;Multiplica por 11
151 velMothership /=10    ;Divide por 10
152 ;Se multiplica velMothership por 1.1
153
154 desplazamientoUsr -    ;Se cambia el signo (inversion de controles)
155
```

- Suma y resta: var +=entero → var = var + entero
- Producto var *=entero → var = var * entero
- Cociente var /=entero → var = var / entero
- Opuesto var - → var = -var

Modificación de asset:

La modificación de un campo de un asset se hace de manera similar a la modificación de un campo de levelSettings, con la diferencia de que hay que crear un bloque al inicio del archivo de nivel con la siguiente forma:








```
1 ASSET_MODIFY 11 ;modifico los aliens
2   shootProb 4
3   maxBullets 10
4   initLives 4
5 END
6 ASSET_MODIFY 1
7   shootProb +=1
8   maxBullets 2
9   initLives 4
10 END
```

Indicando luego de ASSET_MODIFY el número del identificador del asset que se quiere modificar. Es importante que al finalizar el bloque de modificación del asset se encuentre un END. Se aclara nuevamente que la indentación no es necesaria pero aporta mayor prolijidad.

Guía rápida para la creación de niveles

Para crear un nivel se deben seguir los siguientes pasos:

1. En primer lugar, se determinan que assets van a participar y se buscan sus identificadores. Por defecto se encuentran los siguientes Assets:

	ID Linux	ID Raspberry Pi	Tamaño Linux (ancho x alto)	Tamaño RPI	Sprite
Alien1	11	11	54x39	3x2	
Alien2	12	12	59x39	3x2	
Alien3	13	13	54x39	3x2	
Mothership	14	14	90x50	3x1	
Usuario	1	1	39x57	3x2	
Barrera Derecha	21	-	54x76	-	
Barreralzquierda	22	-	54x76	-	
Minibarrera	-	21	-	1x1	

2. Luego se crea un nuevo archivo en la carpeta game/level que se llame de la siguiente manera: plataforma_level_numero.level
 - Siendo plataforma una palabra de 3 letras que identifica la plataforma a la cual pertenece el nivel (lnx | rpi) y el número de nivel.
 - Tampoco es necesario el último y además se puede agregar un texto seguido del número. Un ejemplo de nombre de nivel podría ser lnx_level10.level
3. Al comienzo del archivo de nivel se modifican los assets que se quieran modificar creando los bloques ASSET_MODIFY ID finalizando con un END.
4. En las siguientes líneas, se modifican los campos de levelSettings que se quieran modificar. Un ejemplo sería la cantidad de naves nodrizas que pueden aparecer, valor que debe ser inicializado en cada nivel.

```

1 ASSET_MODIFY 11 ;modifico los aliens1
2   shootProb 4
3   maxBullets 10
4   initLives 4
5 END
6 ASSET_MODIFY 1 ;modifico el usuario
7   maxBullets 2
8   initLives 3
9 END
10
11 maxMShipXLevel 3 ;Seteo la cantidad maxima de naves nodrizas en 3
12 velMothership 20 ;Seteo la velocidad de las naves nodrizas en 20
    
```

5. Luego, se crean los objetos que van a aparecer durante el juego (a excepción de la nave nodriza), dichos objetos se inicializan mediante bloques los cuales deben contener el ID al asset que los representa (utilizando type "numero de ID") y las coordenadas donde van a aparecer.

SPACEINVADERS - Guía para la creación de niveles

6. Se inicializa la posición de la nave del usuario creando un bloque que empiece con USUARIO y termine con END
7. Se inicializan las posiciones de las naves enemigas creando un bloque por cada nave que empiece con ALIEN y termine con END

```
14 ;-----USUARIO-----
15
16 USUARIO
17     type 40
18     posX 100
19     posY 650
20 END
21
22 ;-----ALIENS FILA 1-----
23
24 ALIEN
25     type 1
26     posX 150
27     posY 100
28 END
29
30 ALIEN
31     type 1
32     posX 250
33     posY 100
34 END
```

8. Se inicializan las barreras creando un bloque por cada barrera que empiece con BARRERA y termine con END

```
73 ; Barrera 3
74     BARRERA
75         type 21
76         posX 12
77         posY 11
78     END
79     BARRERA
80         type 21
81         posX 13
82         posY 12
83     END
84     BARRERA
85         type 21
86         posX 12
87         posY 12
88     END
89     BARRERA
90         type 21
91         posX 13
92         posY 11
93     END
```

9. Se pueden agregar comentarios con ;
10. Se guarda el archivo y ya está listo para jugar

NOTA: El orden de inicialización no es importante, la creación de las barreras aliens y usuario se puede realizar en cualquier línea del archivo (a excepción de aquellas que se encuentren dentro de un bloque). Todos los bloques deben finalizar con END. Se recomienda probar el archivo de nivel utilizando por ejemplo el ejecutable de levelTester, el cual indica claramente los pasos a seguir.

Level Tester

En la carpeta raíz se compila utilizando makefile con el siguiente comando:

make levelTester

Luego se ejecuta agregando la carpeta /game utilizando argumentos de entrada. Ejemplo, si se está ejecutando desde la raíz:

./levelTester game