

Informe GitHub

2022
6to 2da
Prof. Albert Ramos
Lucas Hartung
Colegio Tecnológico del Sur

Explicación breve del Juego:

Este es un juego desarrollado en Python, que básicamente consiste en una nave/figura que se mueve sobre los ejes x e y, donde su función principal es desplazarse y disparar, a todo esto, se le suman sonidos de fondo, tanto como de disparo como el de inicio

Puntos a realizar:

1- Realizar una figura:

Para realizar la figura opero dentro de la función if (xpos != bxpos) or (ypos != bypos), donde cambiando los números dentro de las funciones gameState[bxpos+"numero", "numero"+ypos] , variando estos números creo pixeles, y creando muchos pixeles en diferente orden llego a una figura

```
#Borrado y escritura, figura de la nave
if (xpos != bxpos) or (ypos != bypos):

    gameState[bxpos + 5, 59 + bypos] = 0
    gameState[bxpos + 5, 58 + bypos] = 0
    gameState[bxpos + 5, 57 + bypos] = 0
    gameState[bxpos + 5, 56 + bypos] = 0
    gameState[bxpos + 5, 55 + bypos] = 0
    gameState[bxpos + 6, 54 + bypos] = 0
    gameState[bxpos + 6, 59 + bypos] = 0
    gameState[bxpos + 7, 59 + bypos] = 0
    gameState[bxpos + 7, 58 + bypos] = 0
    gameState[bxpos + 7, 57 + bypos] = 0
    gameState[bxpos + 7, 56 + bypos] = 0
    gameState[bxpos + 7, 55 + bypos] = 0

    gameState[xpos+5, 59+ypos] = 1
    gameState[xpos+5, 58+ypos] = 1
    gameState[xpos+5, 57+ypos] = 1
    gameState[xpos+5, 56+ypos] = 1
    gameState[xpos+5, 55+ypos] = 1
    gameState[xpos+6, 54+ypos] = 1

    gameState[xpos+6, 59+ypos] = 1
    gameState[xpos+7, 59+ypos] = 1
    gameState[xpos+7, 58+ypos] = 1
    gameState[xpos+7, 57+ypos] = 1
    gameState[xpos+7, 56+ypos] = 1
    gameState[xpos+7, 55+ypos] = 1
```

2- Cambiar color disparos:

Básicamente para cambiar el color de los disparos, me dirijo donde el gameState es 2 y se cambian los números dentro del segundo paréntesis , estos hacen referencia al estado de intensidad de los colores RGB (Red, Green, Blue) según la intensidad de la combinación de estos 3 colores se logra el color que se quiera obtener

```
#Color de disparos|
elif gameState[x, y] == 2:
    pygame.draw.polygon(screen, (255, 192, 0), poly, 0)
```

3- Cambiar música:

Para cambiar la música me dirijo a la función mixer.music.load y reemplazo el archivo o cambio el nombre al que se desea

```
#Load audio file
mixer.music.load('CoCSong.mp3')
```

4- Añadir sonido a los disparos:

Aquí me dirijo al evento.key pero del espacio la cual es la tecla que ejerce los disparos de la nave, y desde ahí hago el mismo procedimiento que hice para cambiar la música, o cambio el nombre o reemplazo el archivo

```
elif event.key == pygame.K_SPACE:
    gameState[xpos_canon, ypos_canon] = 2
    mixer.music.load('PawPawPaw.mp3')
```

5- Explicar físicamente el movimiento de la nave:

La nave/figura tiene dos tipos de desplazamientos, vertical y horizontal que se basan en el eje x e y, la velocidad de la nave/figura varía según las veces en un segundo que se toque las teclas: “arriba, abajo, derecho o izquierda”

~~6- Evitar que la nave salga de los bordes~~

~~7- Evitar que la nave se destruya~~

8- Cualquier otra modificación o prueba:

Cambio de color a la figura:

Aquí también aplico la misma técnica que la que use para cambiar el color de los disparos, cambio los números RGB, solo que esta vez le cambio a la figura el color

```
#Color de la nave
else:
    pygame.draw.polygon(screen, (0, 0, 255), poly, 0)
```

Cambio de color de celdas vivas:

Aplico la misma técnica de cambiar color que use para la nave y las balas, solo que para las celdas

```
#Celda viva  
pygame.draw.polygon(screen, (20, 250, 243), poly, 0)
```