CC3501: Tarea 1C - Snake

Camilo Núñez Barra

25 de octubre de 2020

Resumen

Una serpiente con hambre infinita de manzanas crece cada vez que come una. La serpiente muere al tocarse a sí misma o al tocar uno de los bordes del mapa en el cual se mueve. Se implementa el patrón de arquitectura de software Modelo–Vista–Controlador, permitiendo al usuario definir el mapa sobre el que la serpiente, modelada como una matriz de tiempos, puede visualizarse en dos dimensiones, mediante la interfaz gráfica de PyOpenGL y GLFW.

1. Solución propuesta

Se utiliza el patrón de arquitectura de software Modelo–Vista–Controlador, de modo que el usuario, al interactuar con la interfaz de GLFW, usa el controlador mediante ciertas teclas, que manipulan los modelos de PyOpenGL, actualizando la vista del usuario y generando así un bucle de interacción que permite la ejecución del juego. El diagrama de este patrón se muestra en la figura 1.

La serpiente tiene un valor propio de tiempo, que cuenta la cantidad de fotogramas ya mostrados, además, la cabeza avanza una casilla en cada fotograma, según la dirección indicada por el controlador. La serpiente se modela como una matriz de tiempos del tamaño de la grilla del mapa, y que se inicializa con valores `-1`. Sea `t` el número actual de fotogramas y `(x,y)` las coordenadas de la posición actual de la cabeza de la serpiente, entonces se asigna `matriz[y][x]=t` para cada iteración. La serpiente tiene largo de valor `L`, que corresponde a la cantidad de manzanas comidas más uno. Luego, solamente se dibujan las partes de la serpiente, mediante texturas, tales que sus coordenadas `(x,y)` cumplen con `t-matriz[y][x]<L`.

Si las coordenadas de la posición de la manzana, que se modela con primitivas de PyOpenGL, coinciden con las coordenadas de la posición de la cabeza de la serpiente, entonces se aumenta el largo en uno y la manzana reaparece en una posición que no contenga una parte de la serpiente.

Si la cabeza tiene una posición fuera de la grilla, y, por lo tanto inválida, la serpiente muere, así mismo si la cabeza coincide con una parte de la serpiente.

2. Instrucciones de ejecución

1. Descargar Python 3.8.5 de https://www.python.org/downloads/release/python-385/ e instalar, añadiendo al PATH.

2. Descargar la carpeta `tarea1c` de https://github.com/camilo-nb/CC3501-tareas. La estructura de esta carpeta se muestra en la figura 2.

3. Abrir el intérprete de comandos en la carpeta `tarea1c`. Por ejemplo, en Microsoft Windows se puede usar `cmd.exe`.

4. Instalar las librerías: `glfw==1.12.0`, `PyOpenGL==3.1.5` y `numpy==1.19.1`, mediante el comando `pip install -r requirements.txt`.

5. Ejecutar el código de la tarea, mediante el comando `python main.py [options]`. La lista completa de comandos de opciones está disponible en la tabla 1. Ingresar (obligatoriamente) el tamaño `N` de la grilla cuadrada con `-n N`. Por ejemplo, se puede ejecutar `python main.py -n 50 -w 1280 -h 720 -f 24 -s off`, para una grilla cuadrada de 50×50, ventana de 1280×720 píxeles de resolución, velocidad de 24 fotogramas por segundos y sonido desactivado.

6. Usar las teclas de movimiento: `W` arriba, `A` izquierda, `S` abajo y `D` derecha, con el objetivo de que la serpiente alcance la manzana. Se pierde al tocar el borde del mapa o al comerse a sí misma, de modo que aparece la escena final, activando las teclas: `F` rendirse y `R` reiniciar.

3. Resultados

La figura 3 muestra la penúltima instrucción de ejecución, que permite crear una ventana, mostrando luego el juego, como, por ejemplo, la ventana que aparece en la figura 4 (los comandos de opciones para ambas figuras son diferentes), que tiene una serpiente de largo 9 y un usuario cuyos últimos tecleos fueron `W`, `A`, y `W`. El usuario eventualmente pierde y se despliega la escena final de la figura 5. El video de demostración se puede ver en https://youtu.be/6lD3HlsB56Q.

Anexo

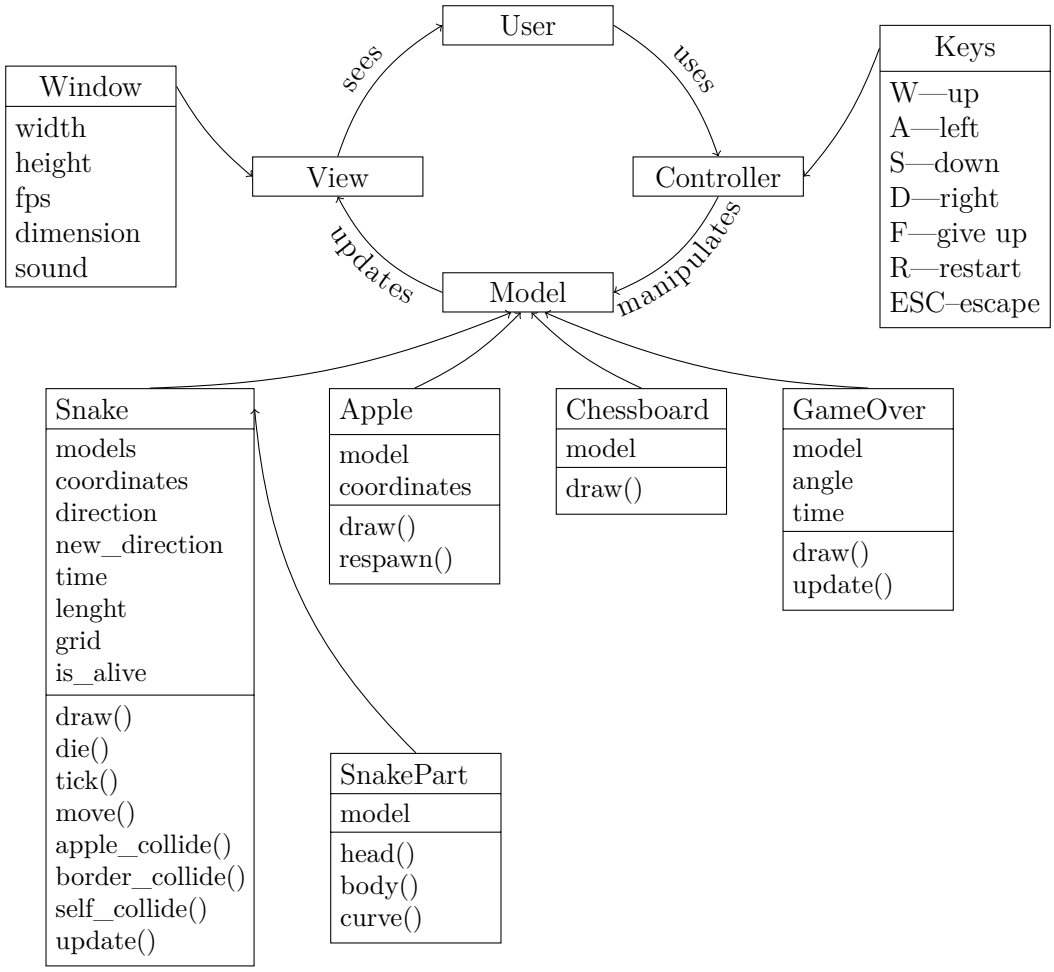

Figura 1: Patrón Modelo–Vista–Controlador

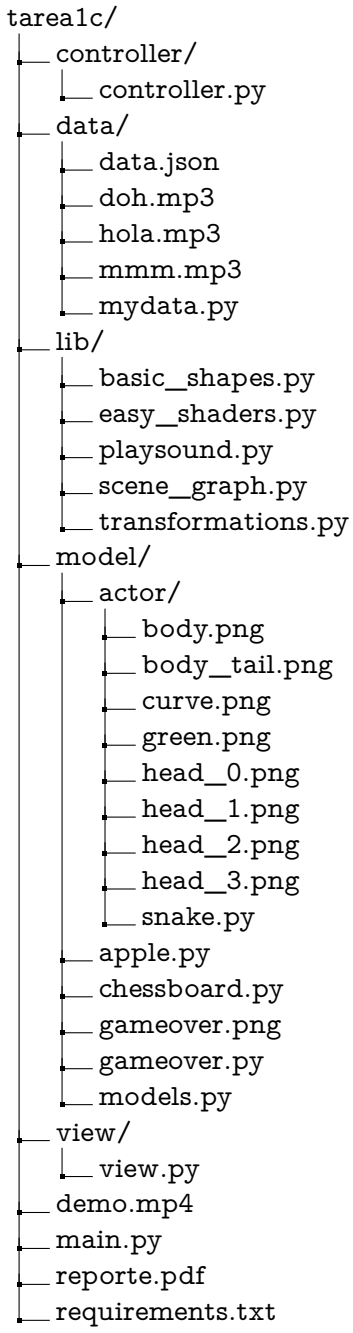

Figura 2: Estructura de la carpeta `tarea1c`

Tabla 1: Comandos de opciones

comando	valor	descripción	tipo	por defecto
`-n`, `--dimension`	int	Dimensiones del mapa (grilla cuadrada)	requerido	—
`-w`, `--width`	int	Ancho de la ventana (pixeles)	opcional	`800`
`-h`, `--height`	int	Alto de la ventana (pixeles)	opcional	`600`
`-f`, `--fps`	int	Fotogramas por segundo (velocidad)	opcional	`10`
`-s`, `--sound`	on/off	Música y efectos especiales	opcional	`on`

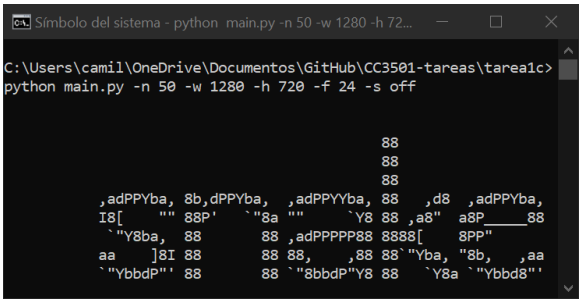

Figura 3: Ejecución del código en el intérprete de comandos

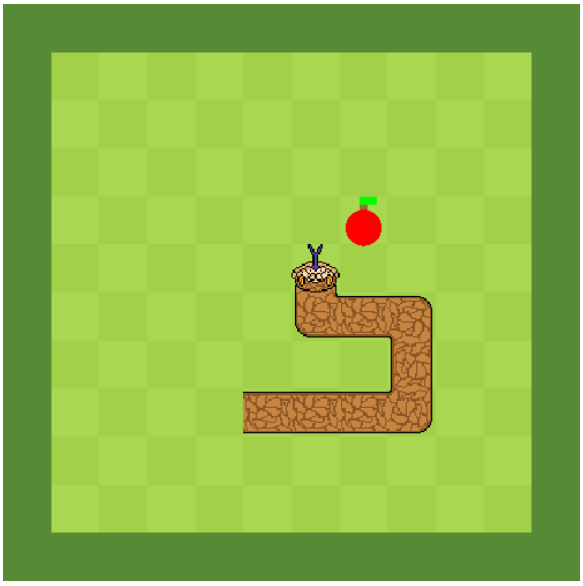

Figura 4: Snake a punto de comer una manzana, en una grilla de tamaño 10

Figura 5: Escena final