

21 | Tetris

Manuel Alcántara Juárez

Prerrequisitos

- Objetos
- Recursividad

Meta

Reafirmar los conocimientos adquiridos durante el curso en los temas de *Arreglos*, *Recursión* y *Creación de Objetos* al desarrollar una aplicación funcional que permita ponerlos en práctica.

Objetivos

Al finalizar la práctica el alumno será capaz de:

- Utilizar objetos de clases definidas por otras personas.
- Definir sus propios tipos de datos para que los utilicen otros programadores.

Antecedentes

Tetris es un videojuego tipo “puzzle” inventado por el ruso Alekséi Pázhitnov el 6 de junio de 1984 cuando estaba trabajando en la Academia de Ciencias de Moscú. Es posiblemente, junto con Pac Man, Super Mario Bros y Space Invaders, el videojuego más conocido de la década de los años 80 y 90. El nombre "tetris" deriva del prefijo "tetra",

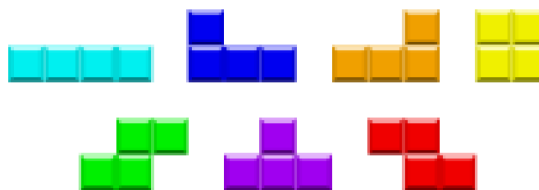


Figura 21.1 Las distintas piezas del Tetris.

que significa "cuatro", y hace referencia a la cantidad de cuadros que componen las piezas. Según la leyenda Alekséi Pázhitnov programó una versión de su juego en una sola tarde.

Mecánica de juego

Distintos tetríminos, que son figuras geométricas compuestas por cuatro bloques cuadrados unidos de forma ortogonal, caen de la parte superior de la pantalla. El jugador no puede impedir esta caída pero puede decidir la rotación de la pieza (0° , 90° , 180° , 270°) y en qué lugar debe caer.

Cuando una línea horizontal se completa, esa línea desaparece y todas las piezas que están por encima descienden una posición, liberando espacio de juego y por tanto facilitando la tarea de situar nuevas piezas. La caída de las piezas se acelera progresivamente. El juego acaba cuando las piezas se amontonan hasta salir del área de juego.

Existen distintas versiones del juego. La original tiene siete piezas diferentes, licencias posteriores añadieron formas suplementarias y existen incluso ciertas licencias para formas tridimensionales.

Desarrollo

Tu misión en este proyecto, es simular una versión del juego del Tetris, que cuente solo con las funciones básicas del juego original, es decir, que se puedan rotar las piezas y que se lleve la cuenta del marcador.

Ejercicios

1. Clona tu proyecto y ejecútalo utilizando Ant. La primera vez verás la siguiente pantalla de la Figura 21.2(a).
2. Ahora, lo primero que tienes que hacer es completar la clase **Point2D**, en especial los métodos marcados con el comentario: `//Completar`

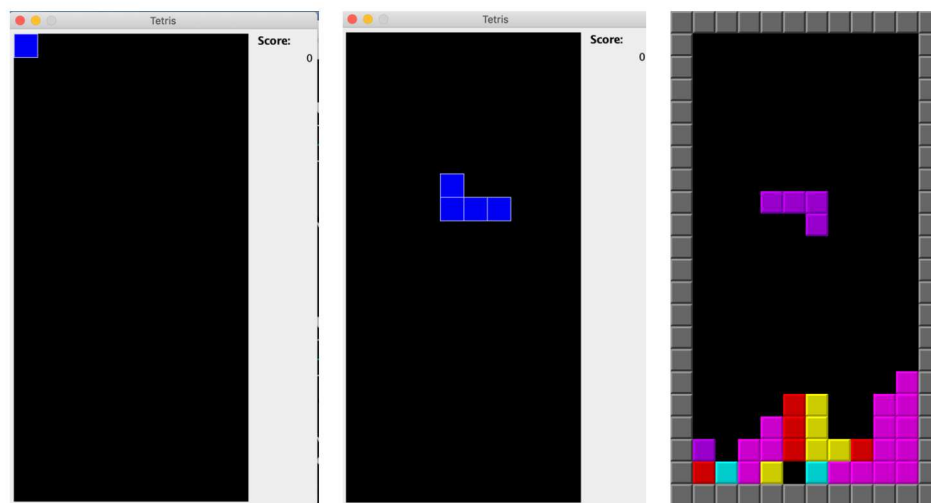


Figura 21.2 (a) Lo que muestra el código auxiliar al inicio. (b) Con las piezas programadas. (c) Tetris.

3. Una vez hayas terminado, cuando vuelvas a ejecutar tu proyecto debes de ver una pieza que va bajando como en la Figura 21.2(b).
4. Usando como Plantilla la clase **Tetrimino2**. Crea clases para los Tetrimonos faltantes **Tetrimino1**, **Tetrimino3** ... **Tetrimino7**.
5. Cuando hayas finalizado, agrega tus figuras en la clase **Tetris.java**. El comentario: `//Insertar figuras aquí`, te dará una pista.
6. Si todo sale bien, verás que van a ir cayendo piezas de manera aleatoria como en la Figura 21.2(c).
7. Un detalle con el que te habrás dado cuenta, es que una vez que completas una línea esta no desaparecerá. Por eso, tienes que implementar esa parte. Ve a la clase **Board.java** e implementa el método que se llama **isLastRowFilled**, el cual tiene que regresar true si el último renglón del juego esta lleno o false en otro caso.
8. Utiliza esa función dentro del método **deleteFilledRows** para checar si la última hilera esta llena, y de ser así tienes que copiar todos los renglones del tablero un renglón hacia abajo. El método tiene que regresar cuántos renglones se eliminaron. Debido a que hay ocasiones en donde una sola pieza puede eliminar varios renglones, tienes que hacer esta función recursiva. Pero si gustas primero pruébala sin usar recursión para ver que al menos la ultima hilera se elimina si se completa.
9. Felicidades has completado tu misión.
10. (Opcional) Ve duplicando el número de puntos de acuerdo a la cantidad de renglones eliminados en una sola pasada. Si solo se elimina 1 entonces se otorgará

- 1 punto, si se eliminan 2 entonces 2, si se eliminan 3 tendría que ser 4, para 4 reglones los puntos a sumar serían 8 y así sucesivamente.
11. (Opcional) Modifica el evento cuando se usa la tecla DOWN para que en vez de girar el tetrimino a la izquierda haga que baje de un jalón hasta que tope con el borde o con otra figura.