

Instituto Tecnológico de Costa Rica

Algoritmos y Estructuras de Datos I

I Proyecto Programado

Tron

Josue Venegas Arias 2023075124

Leonardo Andrés Araya Martínez

8 de septiembre de 2024

Introducción

El presente documento corresponde a la documentación del proyecto "Tron", desarrollado como parte del curso de Algoritmos y Estructuras de Datos II en el Instituto Tecnológico de Costa Rica. El objetivo principal es implementar un juego de carreras de motos de luz, utilizando estructuras de datos lineales como listas enlazadas, pilas y colas. Este proyecto está diseñado para aplicar diversos conceptos de estructuras de datos, algoritmos y patrones de diseño en la resolución de un problema específico.

En este documento se detallan los diferentes componentes del proyecto, incluyendo las decisiones de diseño tomadas, la implementación de las estructuras de datos, la integración de los elementos gráficos en C# con MonoGame.

Objetivos del proyecto

El proyecto "Tron" tiene como objetivo principal implementar un juego de carreras de motos de luz, utilizando estructuras de datos lineales para gestionar el movimiento, la estela, y la interacción entre las motos y el entorno.

- **Implementar las diferentes estructuras de datos lineales:** Esto incluye listas enlazadas, pilas y colas, que son necesarias para modelar los comportamientos clave del juego.
- **Desarrollar algoritmos para gestionar el movimiento de las motos:** Los algoritmos deben garantizar que las motos puedan moverse de manera continua en un grid, evitando colisiones y gestionando la longitud de su estela.
- **Integrar las estructuras de datos con la interfaz gráfica:** Esto implica sincronizar las actualizaciones en la estructura de datos con las actualizaciones gráficas en la pantalla del juego, para asegurar que el jugador vea los cambios en tiempo real.

Descripción del Problema

El juego de "Tron" consiste en un entorno de carreras de motos de luz, donde los jugadores controlan una moto que deja una estela destructiva a su paso. El objetivo es que las motos puedan moverse en un mapa en forma de grid, y que si una moto cruza la estela de otra, o choca con una pared, esta se destruye. Además, se deben considerar factores como la velocidad, el combustible y la longitud de la estela, los cuales afectan directamente el comportamiento de cada moto.

El reto principal del proyecto es implementar las motos utilizando estructuras de datos eficientes, que permitan representar tanto el movimiento como la estela de cada una, y hacer que estas estructuras interactúen correctamente con el mapa.

La solución desarrollada se basa en el uso de una lista enlazada simple para representar cada moto y su estela. La cabeza de la lista corresponde a la posición actual de la moto, mientras que los nodos sucesivos representan las posiciones anteriores (la estela). Conforme la moto avanza en el grid, el nodo que antes representaba la última posición se elimina de la lista y se agrega un nuevo nodo en la cabeza, siguiendo la dirección del movimiento de la moto.

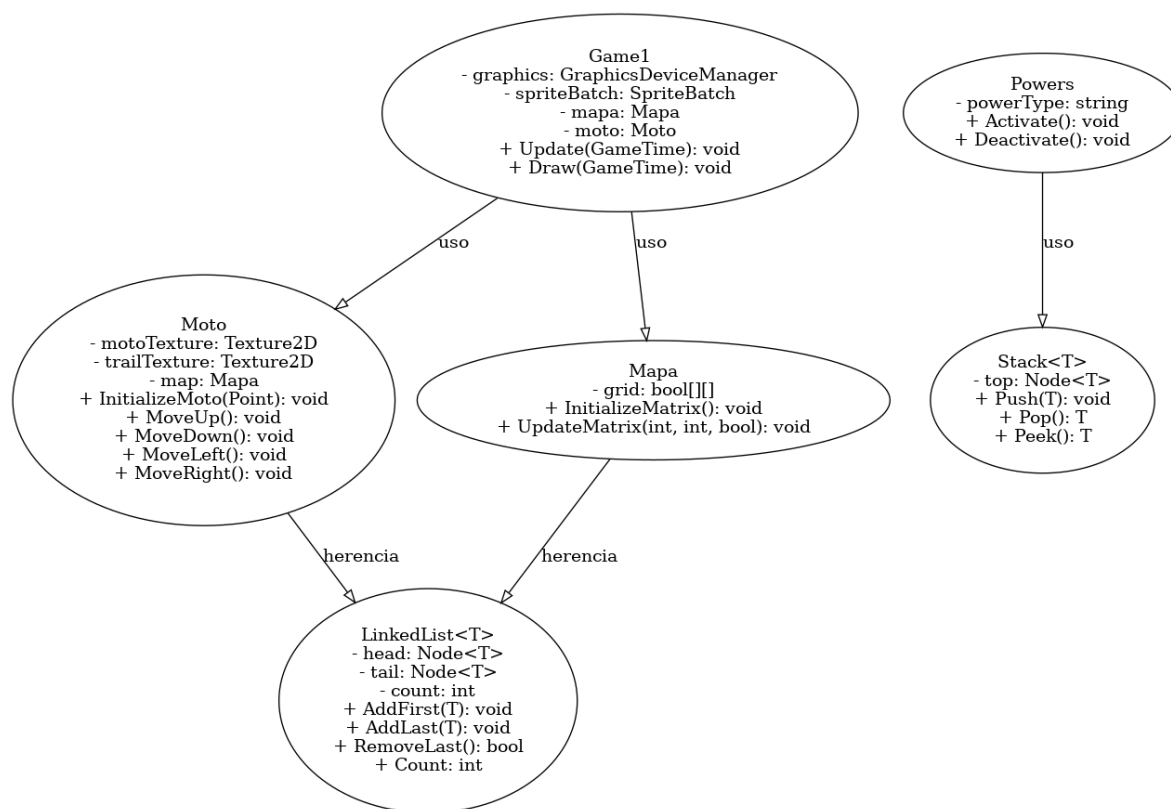
Cada vez que la moto se mueve, se actualiza la matriz (que también se implementa como una lista enlazada) y la interfaz gráfica, mostrando las posiciones actuales de la moto y su estela. Además, se desarrollaron métodos que permiten el movimiento de las motos en las cuatro direcciones cardinales, y se implementó una lógica para manejar la destrucción de la moto al chocar con estelas o los límites del mapa, y a su vez la implementación de habilidades (gadgets) que se acumulan en stack.

Problemas Encontrados y Soluciones

Durante el desarrollo del proyecto se encontraron algunos problemas, especialmente relacionados con el manejo de las posiciones y las colisiones. Uno de los retos fue asegurarse de que las motos no salieran del área visible de la pantalla, lo cual se solucionó ajustando las coordenadas iniciales y aplicando límites al movimiento. Además, se implementó una lógica básica para detectar colisiones entre las motos y las estelas.

Otro desafío fue la integración correcta de las estructuras de datos con la interfaz gráfica, para lo cual se optimizaron los métodos de actualización de la matriz y el redibujado de las motos.

Diagrama de Clases



Link GitHub

<https://github.com/Cotue/TronGame.git>