Project Proposal

Project Description

The name of my term project is Rail Control. This is a traffic control game where the player needs to control the rails to ensure that the cars can safely reach their destination. The game will start with a random map of different types of rails and destinations. The player will interact with the game by changing the direction of the rails by clicking them. Different types of cars will spawn from the edge of the map. The player will win points if a car successfully reaches its corresponding destination and lose points if two cars crash into each other.

Similar projects

The goal of my term project game will be similar to Mini Motorways in that in both games we want to have the cars going from one place to another. However, what is different is that the rails/roads will be predetermined in my term project, and the player will instead interact with the game by switching the direction of the rails instead of making new rails.

The graphics of my term project will be similar to Railbound in the ways that it will be grid based and include different kinds of rails. What is different in my term project is that it will be 2D Bird's-Eye View and the graphics might only be simple shapes.

Structural Plan

I will have one class in each file. Most of my functions will be inside classes, with the exception of the main file that runs the animation and a few general helper functions, such as one that will be used to calculate distance.

Classes I will need (not a exhaustive list of functions):

- Map
  - Includes the algorithm to generate the random map
  - Store all rails, destinations, and cars
  - Function to display all components of the map on screen by looping over all the stored objects and calling their display functions
  - Function to check if any car reached a destination or any cars crash
- Rail
  - Attributes that store its position, orientation, and destinations it is next to
  - A function to display the rail on screen
  - A function to change orientation if possible when the mouse clicks on it
- Car
  - Attributes that store its position and type (which determine its color, speed, etc)
  - A function to display the car on screen
  - A function to move the car from rail to rail
- Destination
  - Attributes that store its position and type
  - A function to display the destination on screen

Algorithmic Plan

The algorithmically most difficult part of my project is to randomly generate the maps with rails and destinations. I plan to use the DFS algorithm that generates a maze to help me generate the map of rails. Then I am probably just going to place the destinations randomly towards the center of the map for the MVP and possibly find an improvement if I have time.

Timeline Plan

TP1

- Random generation of the map
- Functioning rails which directions can be changed upon click
- Spawning of cars that will move on rails

TP2

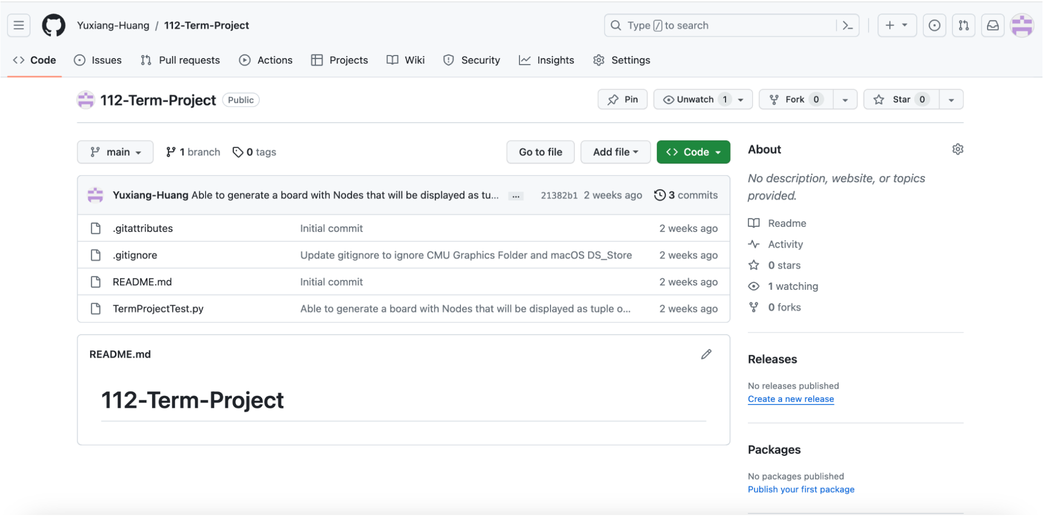
- Placement of destinations
- Functioning destinations

TP3

- Polish the game, including adding UI and better graphics
- Add more complex rails

Version Control Plan

I will use a Github repository for version control to back up my code.



Yuxiang-Huang Able to generate a board with Nodes that will be displayed as tu... 21382b1 2 weeks ago 3 commits

.gitattributes	Initial commit	2 weeks ago
.gitignore	Update gitignore to ignore CMU Graphics Folder and macOS DS_Store	2 weeks ago
README.md	Initial commit	2 weeks ago
TermProjectTest.py	Able to generate a board with Nodes that will be displayed as tuple o...	2 weeks ago

README.md

112-Term-Project

No description, website, or topics provided.

No releases published

No packages published

Module List

I am not planning on using any additional modules.

TP1 Update

I made no significant change. The only “change” is that a rail can have at most one car on it at a time, so the demonstration in the storyboard wouldn’t be valid. This is less of a change than a mistake I made when drawing the storyboard…

TP2 Update

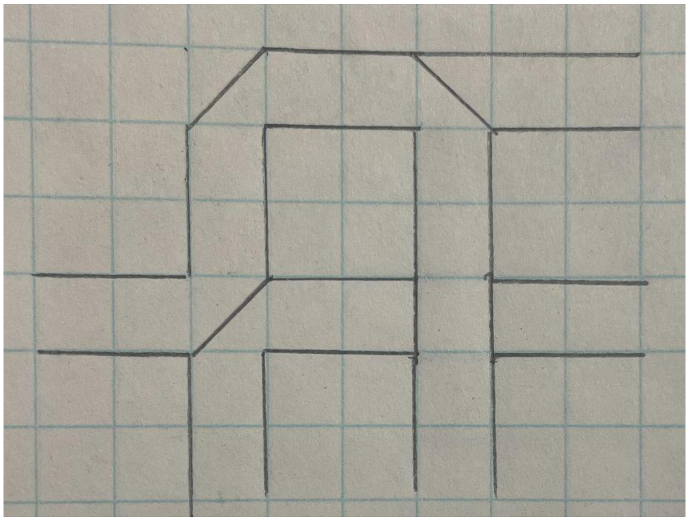
No changes were made.

TP3 Update

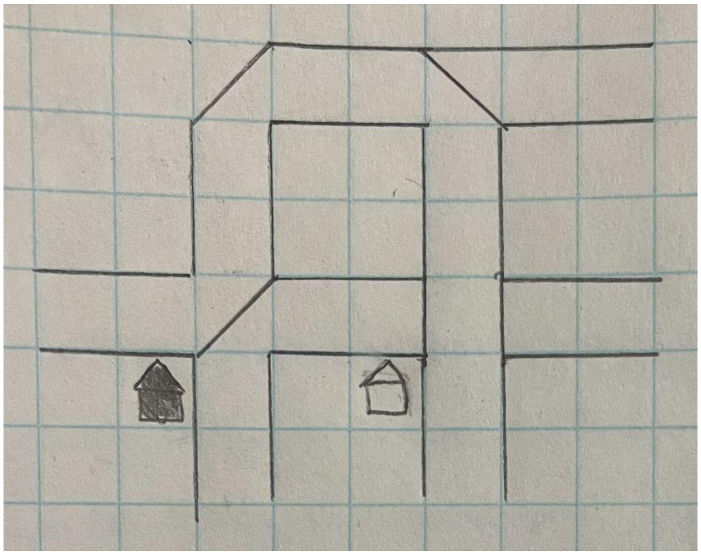
No changes were made.

Storyboard

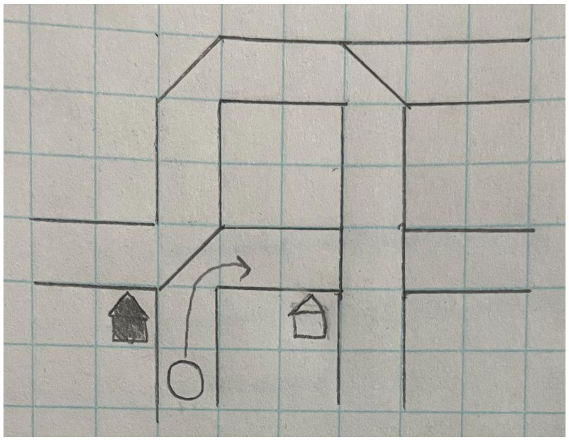
A map with rails. (This is a very small map; the actual map will include many more rails)



A map with rails and two types of destinations.

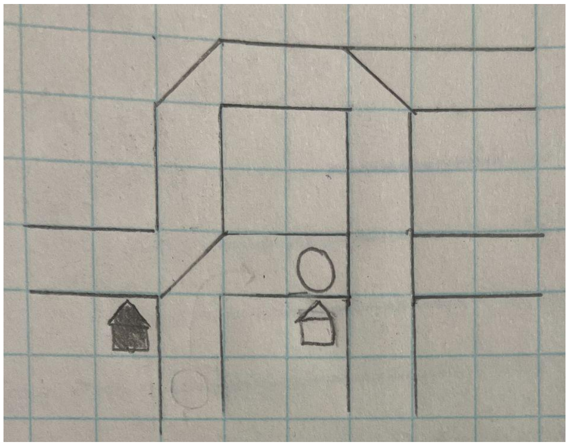


A car spawned from the edge of the map! (move according to the rail)

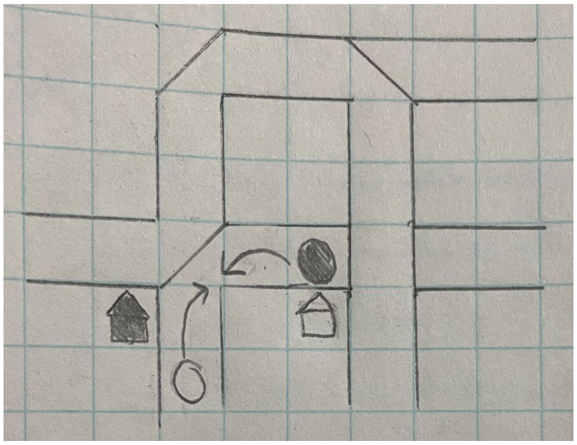


The player wins points when a car reaches its corresponding destination!

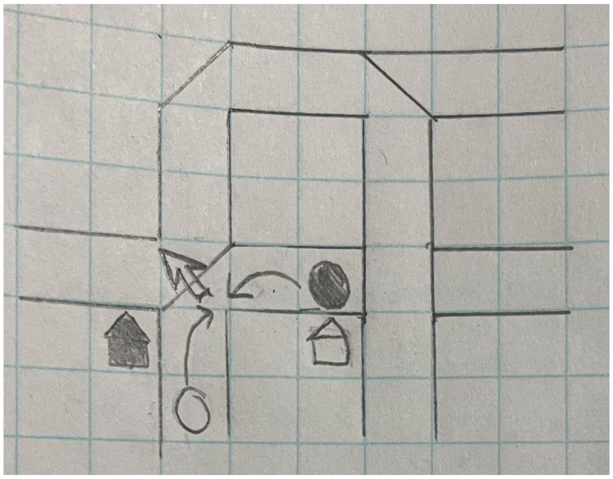
(Also car just stops when it is blocked)

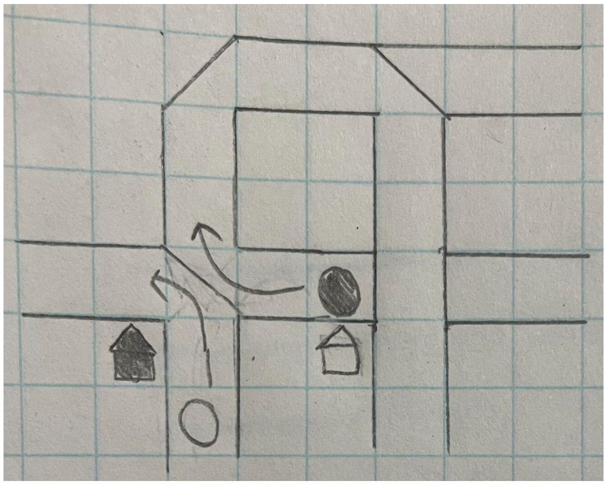


The player will lose points if two cars crash :(

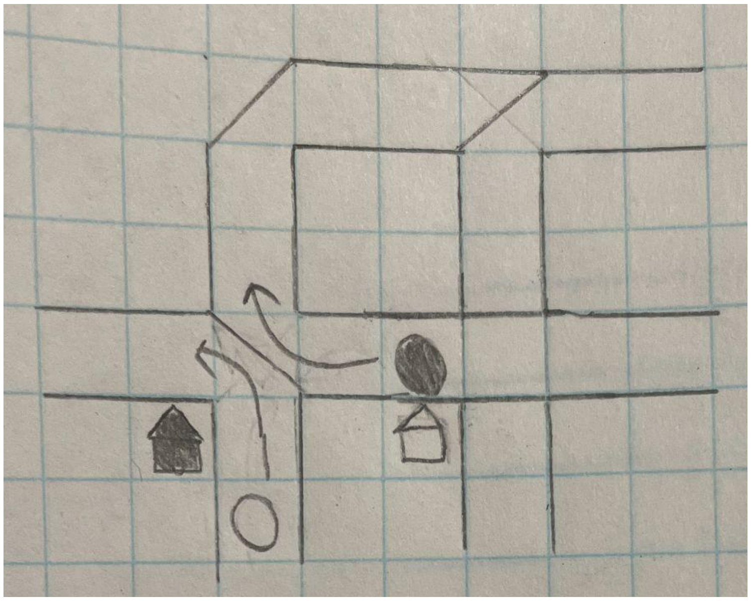


Click on the rail to change its destination to prevent the two cars from crashing!





Demonstration of how the other rails change when clicked



Preliminary Code Files are also included in this zip file !!!

(It is the very first rough attempt to create the map of rails)