

# COSC363: Assignment 1

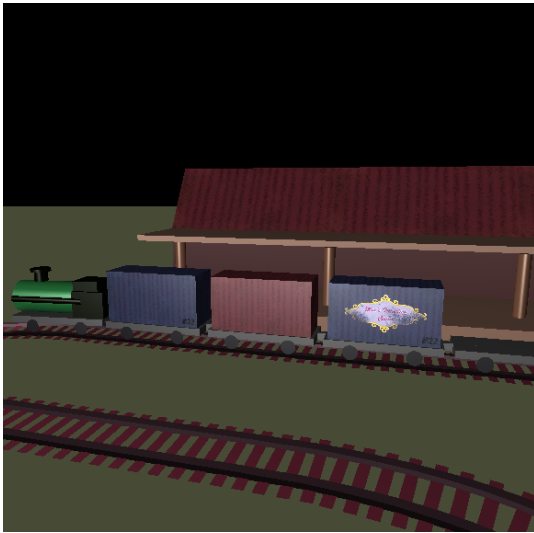
## Report

Max Bastida

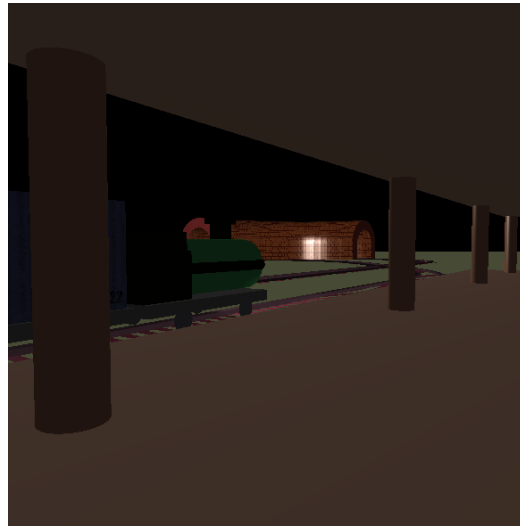
89476639

### Description.

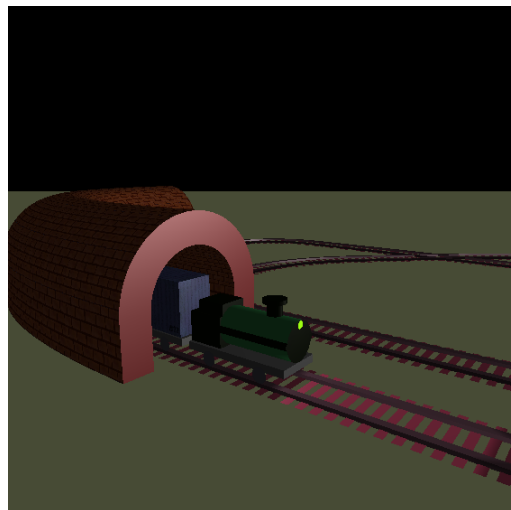
The scene shows a train on a looping track. The train has a green engine that pulls a series of red and blue containers (Fig 1). The track is shaped in two loops, such that there is a smaller loop inside of a larger one. Along a straight edge of the outside loop of the track is a train station. During every loop of the track, the train stops for a while at the station (Fig 2). On a curve of the outside loop of the track is an arch shaped brick tunnel, which the train tracks pass through (Fig 3).



**Fig 1:** The model train and station.



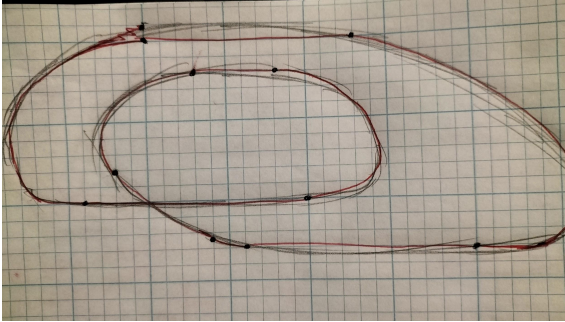
**Fig 2:** The train station.



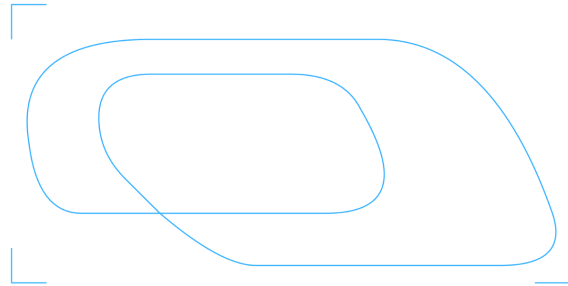
**Fig 3:** The tunnel

### Extra features: Non-circular track

The track was modelled by reading in a csv file of points on the track, and using these to generate a track model. The track design was created first by sketching on graph paper (Fig 4), and then using Virtual Graph Paper<sup>1</sup> to create a clean version (Fig 5).



**Fig 4:** Track design sketch



**Fig 5:** Graph from Virtual Graph Paper

The track design from Virtual Graph Paper was then inputted into Web Plot Digitizer<sup>2</sup>, which generated a csv file of points on the track. Edits were then made to this csv file to correct errors that came from points being too far apart and breaks in the track from the place where the track crosses itself.

The track was modelled using the method described in the lecture notes, where for each track segment, three consecutive points (P1, P2 and P3) along the track would be taken from the csv file. Unit vectors orthogonal to the directions P1 to P2 and P2 to P3 would be found, and used to plot quadrilaterals that formed the track segments. For a flat track segment with corners at coordinates C1, C2, C3 and C4, the coordinates would be calculated as follows:

$C1 = (P1x + V1x * W1, 0, P1z + V1z * W1)$ ;  $C2 = (P1x + V1x * W2, 0, P1z + V1z * W2)$ ;  
 $C3 = (P2x + V2x * W2, 0, P2z + V2z * W2)$ ;  $C4 = (P2x + V2x * W2, 0, P2z + V2z * W2)$ ;

where V1 is the unit vector perpendicular to the vector between P1 and P2, V2 is the unit vector perpendicular to the vector between P2 and P3, and W1 and W2 are the distances each point is from the centre of the track. W1 is the distance from the centre of the inner edge, or half the width of the track, and W2 is (W1 + the width of a single rail).

To model the second rail, this process would be repeated with  $W1 = W1 * -1$  and  $W2 = W2 * -1$ .

<sup>1</sup> Virtual Graph Paper <https://virtual-graph-paper.com>

<sup>2</sup> Web Plot Digitizer <https://automeris.io/WebPlotDigitizer>

Positioning the train and tunnel on the non circular track was done with a similar process of finding the unit direction vectors between consecutive points and using this to model the rotation of objects placed around the track.

### **Extra Features: Model enhancements**

The appearance of the track was improved by adding sleepers. This was done using the method described in the lectures, where using the previously calculated  $V1$ , a quadrilateral would be created that had corners at positions  $P1 + V1 * W3$ ,  $P1 + V1 * W3 + U1 * W4$ ,  $P1 - V1 * W3 + U1 * W4$  and  $P1 - V1 * W3$ , where  $U1$  is the unit vector from  $P1$  to  $P2$ , and  $W3$  and  $W4$  are the length and width of the sleepers, respectively.

The appearance of the train and tunnel were enhanced by adding textures. The brick texture for the tunnel was personally drawn, and the textures for the train carriages were created for the train by Georgia Holland<sup>3</sup>.

The design of the train was modified slightly from the lab 2 design to change the size and colours. A chimney was added to the engine, as well as bars along the side to make the design more interesting.

### **Extra Features: Alternate camera views**

Two alternate camera views were added, one at the train station and one attached to the front of the train. The position of the camera at the front of the train was mapped by using the points and vectors found the same way as describe previously. The position of the camera was calculated using  $P1$  translated in the direction of  $U1$  to move it to the front of the train. The look point of the camera was calculated in the same way, with a larger translation, to keep the camera looking the way the train was facing.

### **Camera controls**

Move main camera with:

Up arrow:	move camera forwards
Down arrow:	move camera backwards
Left arrow:	turn camera to the left
Right arrow:	turn camera to the right
Page up:	move camera upwards
Page down:	move camera downwards

Toggle between alternate views with the 'c' key.

The three available camera views are the moveable main camera, the station view, and the cab view, which moves with the train.

---

<sup>3</sup> 'train\_car\_blue', 'train\_car\_bluem', 'train\_car\_logo' and 'train\_car\_red' textures by Georgia Holland

## **Instructions**

To compile and run the program with CLion:

Open the program folder in CLion.

When the folder is opened, it should detect the CMakeLists file and load the project. If this doesn't happen, right click on the CMakeLists file and select Load CMake Project.

Go to the Run menu and select Edit Configurations.

Select the RailwayWorld.out configuration and set the working directory to be the program folder. Apply and close the configurations menu.

Select Run.

## **References**

Textures 'train\_car\_blue', 'train\_car\_bluem', 'train\_car\_logo' and 'train\_car\_red' created by Georgia Holland.

Track shape created using Virtual Graph Paper

<https://virtual-graph-paper.com>

Accessed 5/4/2022.

Track converted into points using Web Plot Digitizer

<https://automeris.io/WebPlotDigitizer>

Accessed 6/4/2022