Documentation:

1. ***All feature I did***

|  |  |
| --- | --- |
| Basic feature(all) | How to see it |
| Have a track that is (at least) C1/G1 | You can choose the Cardinal Cubic. |
| Your track should interpolate (when using interpolating curves) or approximate the control points. | Yes, it is. You can add control point to check. |
| Have the train oriented correctly on the track. | Yes. Please run the train to check. |
| Allow the user to "ride" the train (look out from the front of the train). | Yes, click the “Train” button to get train camera. |
| Have some scenery in the world besides the ground plane. | I changed the ground color and sky color, and add some trees. |
| Your program is properly documented, is turned in correctly, and has sufficient instructions on how to use it in the readme file. | Here is it. |
| You should have a slider (or some control) that allows for the speed of the train to be adjusted (how far the train goes on each step, not the number of steps per second). | You can change the speed by changing the speed slider. |
|  |  |
| Advanced feature(partial) |  |
| Arc-Length Parameterization (BIG) | Click the Arc-Length button. |
| Approximating C2 Curves (little work, but valuable)) | Choose the cubic B-Spline. |
| Draw nicer looking tracks (medium) | You can switch between line, wood track and road. |
| Implement simple physics (medium) | Click the “physics” button to see the change |
| Bells and Whistles  (partial) |  |
| If you’re really into trains, you could have different kinds of cars. | You can switch between tank and train. (I hope you believe it is a tank, anyway I believe so)  However, there is something not right with the “train” model so when you use the train instead of the tank, it may not runs  Consecutively. Fortunately, the “tank” works very well. |

***2. An explanation of the types of curves you have created.***

There are three kinds of curves:

C(0): Linear; Just use line between two control points. Using a for loop goes for all of the control points so that the track ends with the start.

C(1):Cardinal-Cubic; For this spline, I use a matrix to interpolate the point.

C(2):Cubic B-Spline.; Also a matrix, but different from the spline above.

***3. A discussion of any important, technical details.***

For the coordinate system for the train, I just translate the train to the center (0, 0, 0) and then rotate it, then translate it to the place it should go.

For the arc-length, I just use a parameter to accumulate calculate the distance of the next step. If the accumulated distance is bigger than some constant, then I will let the train move. Otherwise the train won’t move until the distance accumulated bigger than that constant.

For C1 and C2 spine, I use the matrix to calculate small point. The parameters increase from 0 to 1 by 0.01 each time. It is possible that if you stretch the track the line can be distinguishable points. That’s why I made the wooden-like track.

***4. Any non-standard changes that you make to the code.***

I didn’t make any changes to the framework.

***5. Anything else we should know to compile and use your program***

I put my own track files in the folder called “My TrackFiles”.

It runs correctly on the lab computers, so I don’t think we need anything else.

Screen Shots:













