Traffic Simulation

Requirements and Test Document

Jack Lee

11 November 2022

CS 225, Fall 2022

Embry-Riddle Aeronautical University

Daytona Beach campus

1 Aerospace Boulevard

Daytona Beach, FL 32114

**Introduction:**

The Traffic Simulation simulates different Cars and Drivers and how they react to changes in driving conditions. Cars and Drivers are objects of GenericCar and GenericDriver respectively, but they are referred to simply as Cars and Drivers in this document. These changes include inclement weather, traffic jams, speed limit changes, and lane changes. The road is straight and 100 miles long. Once a Car reaches the end of the road, its position resets to 0 and it keeps travelling. The simulation uses file input to determine which cars start at which positions on the road and simulates them once they are created in the proper places.

**Background information:**

The simulation reports three things to the user: Total Cars, Average Speed, and Traffic Flow Rate. The total cars can be calculated by adding the number of Cars on the road. The average speed is equal to the sum of all Car speeds divided by the number of total cars. The traffic flow rate is the product of the average speed and the traffic density (University of Idaho, 2003). Traffic density is equal to the number of cars within a distance range, which can be defined by the user. Car position is stored as a double array, with the first index used to store the Car’s distance from 0 and the second index used to store the Car’s lane. Since the simulation is designed to interpret file input, the files must be structured in a specific way. Traffic Simulation files are written in the following format: “Car type: number of Cars, Driver type, position.”

The simulation can be paused while running by pressing the “p” key. While paused, the pause menu appears. The pause menu has different options, including Change Speed Limit, Change Weather, Add/Remove Lanes, Slow Down Lane, and Play. These options load subsequent menus that allow the user to access the described options.

**Requirements:**

Table 1 contains the requirements for the simulation. The requirements are associated with the test cases, and each requirement is based on a user story. The test cases correspond to specific requirements identified by the Requirement ID. This is to identify the requirement the test case seeks to verify.

Table 1: Requirement Specifications

|  |  |
| --- | --- |
| **Reqt**  **ID** | **Requirement Specification** |
| As a software designer, I want to assign different Drivers to Cars. | |
| 1 | Car objects contain a Driver attribute that can be modified and retrieved. |
| As a software designer, I want to put Cars on the road. | |
| 2 | Car objects have a position attribute that can change depending on the Car’s behavior. |
| As a user, I want to see statistics about the Cars. | |
| 3 | Total Cars is displayed to the user. |
| 3.1 | Average Speed is displayed to the user. |
| 3.2 | Traffic Flow Rate is displayed to the user. |
| As a software designer, I want the Cars to respond to slowdowns and traffic. | |
| 4 | Cars will decelerate when another car appears in their visibility range. |
| As a software designer, I want the cars to avoid accidents. | |
| 5 | Cars will only change lanes when the space next to them is open. |
| As a user, I want to simulate text files. | |
| 6 | Traffic files are interpreted and loaded by the program. |
| As a user, I want to know if files I load have errors. | |
| 7 | Files that fail to load will report an error to the user and give the option to try again. |
| As a user, I want to pause the simulation. | |
| 8 | The simulation can be paused at any time and the Cars will stop moving. |
| As a user, I want to play the simulation once paused. | |
| 9 | A paused simulation can be played, and the cars will keep moving from where they left off. |
| As a user, I want to be able to change and define the speed limit. | |
| 10 | The speed limit can change according to the user. |
| As a user, I want to change the weather and lane number. | |
| 11 | The weather can be changed by the user. |
| 11.1 | The lane number can be changed by the user. |
| As a user, I want to slow down Cars. | |
| 12 | Cars in a chosen lane will slow down if selected. |

**Test Cases:**

Table 2 contains the test cases and their results. It will be completed with each sprint as functionality is added to the simulation. Each test case is related to a Requirement, as denoted by the Requirement ID column. Also included are columns for the input, expected output, actual output, and whether the test passed or failed.

Table 2: Test Cases and Results

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Req’t**  **ID** | **Test**  **Case**  **ID** | **Initial**  **Conditions**  **And Input** | **Expected Behavior**  **Or Output** | **Actual**  **Behavior**  **Or Output** | **Pass**  **Fail** |
| 1 | 1 | Build a Car object with a Driver object as a constructor parameter | Code compiles without errors | Code compiled without errors | Pass |
| 1 | 2 | Use a Car’s getDriver() method to run Driver methods | Driver methods run and code compiles | Driver method getMaxSpeed runs, code compiles | Pass |
| 1 | 3 | Print a Car’s getDriverInfo() method | The Driver’s toString() is printed | Driver’s toString() is printed | Pass |
| 2 | 4 | Print a car’s position before and after using setPosition() to change the distance | The distance is equal to the new value when it is printed | Distance changes after setPosition is used | Pass |
| 2 | 5 | Print a car’s position before and after using setPosition() to change the lane | The lane changes to the set value | The lane changes to the set value | Pass |
| 2 | 6 | Use setSpeed() and record the distance over time | The distance changes by the speed | The distance changes by the speed | Pass |
| 3 | 7 | Run the simulation | Total Cars is printed to the console | Total Cars is printed to the console | Pass |
| 3.1 | 8 | Run the simulation | Average Speed is printed to the console | Average Speed is printed to the console | Pass |
| 3.2 | 9 | Run the simulation | Traffic Flow Rate is printed to the console | Traffic Flow Rate is printed to the console | Pass |
| 4 | 10 | Place a Car inside a test Car’s visibility range | The test Car decelerates | The test Car decelerated | Pass |
| 4 | 11 | Place a Car outside a test Car’s visibility range | The test Car does not decelerate | The test Car does not decelerate | Pass |
| 4 | 12 | The user selects a lane to slow down | The leading Car decelerates to a stop | N/A for P5  (Lane Slowdowns implemented later) | N/A |
| 5 | 13 | Place a Car at the same distance in an adjacent lane to another Car | isNextLaneOpen returns -1 | isNextLaneOpen returns -1 | Pass |
| 5 | 14 | Place a Car at a distance where there are no adjacent Cars in other lanes and slow down the leading Car | The Car will change lanes | The test Car changes lanes | Pass |
| 5 | 15 | Place a Car at a distance where there are no adjacent Cars in other lanes | The car will stay in its lane | The test Car stays in the same lane | Pass |
| 6 | 16 | A .txt file in the correct format is loaded | No error, simulation runs |  |  |
| 6 | 17 | A .jpeg file in the correct format is loaded | No error, simulation runs |  |  |
| 6 | 18 | A .traffic file in the correct format is loaded | No error, simulation runs |  |  |
| 7 | 19 | The Car Type field is missing | “Error: This file cannot be interpreted. “ |  |  |
| 7 | 20 | The file ends without a period | “Error: This file cannot be interpreted. “ |  |  |
| 7 | 21 | The file is in the correct format | No error, simulation runs |  |  |
| 8 | 22 | The “p” key is pressed during the simulation | The simulation pauses and an option menu prints |  |  |
| 8 | 23 | No keys are pressed during the simulation | The simulation continues to run |  |  |
| 8 | 24 | Any other key than “p” is pressed during the simulation | The simulation continues to run |  |  |
| 9 | 25 | The word “play” is entered to the console when the simulation is paused | The simulation resumes |  |  |
| 9 | 26 | The key “p” is pressed when the simulation is paused | Nothing happens |  |  |
| 9 | 27 | Any other key is pressed when the simulation is paused | Nothing happens |  |  |
| 10 | 28 | In the speed limit menu, the user enters a number less than the speed limit | The speed limit changes to the entered number |  |  |
| 10 | 29 | In the speed limit menu, the user enters a number greater than the speed limit | The speed limit changes to the entered number |  |  |
| 10 | 30 | In the speed limit menu, the user enters a number equal to the speed limit | The speed limit variable changes to the new value, but the value remains the same |  |  |
| 11 | 31 | In the weather menu, the weather is changed | The weather is set to the new value |  |  |
| 11.1 | 32 | In the Add/Remove Lanes menu, a lane is removed | If more than one lane exists, the lane is removed and all cars in it are removed from the simulation |  |  |
| 11.1 | 33 | In the Add/Remove Lanes menu, a lane is added | A lane is added in the simulation |  |  |
| 12 | 34 | In the Slow Down Lane menu, a lane is selected to slow down to a specified speed lower than the speed limit | The leading car in the lane begins to decelerate until reaching the specified speed |  |  |
| 12 | 35 | In the Slow Down Lane menu, a lane that does not exist is selected to slow down | “Error: That lane does not exist” |  |  |
| 12 | 36 | In the Slow Down Lane menu, a lane is selected to slow down to a speed greater than the speed limit | “Error: That speed is outside the speed limit” |  |  |

**References:**

University of Idaho. (2003). *Speed-Flow-Density relationship*. Transportation Engineering Online Lab Manual; University of Idaho. <https://www.webpages.uidaho.edu/niatt_labmanual/Chapters/trafficflowtheory/theoryandconcepts/SpeedFlowDensityRelationship.htm>

**Appendices:**