G4RCE Development: Design Documentation

Feature: 1
Name: Interactable Game Menu

Description: User will be able to click on buttons and be brought into the next menu or into a race, depending on what they want to do. There are currently four buttons: Play, Track, Garage, Options, and Quit. Near each button will be a view of the current selections, along with what is selected being clearly indicated in the submenus. Each submenu will initially have defaults selected.

Design Details:

Play Button: Pressing this button will bring the user directly into a race. The track and car settings in that race will depend on what is selected in the track, garage, and options submenus.

Track Button: Pressing this button will bring the user into a screen where they will be shown premade tracks to select from, an option to make their own track, and any other options having to do with tracks that may later be added to the game.

Garage Button: Pressing this button will bring the user into a menu that will eventually provide options for car color, number of cars, etc., and an option to enter a submenu where they will be able to enter code for how their car(s) will move. If the user does not enter code, a WASD mode will stay selected.

Options Button: This will lead to a screen that will eventually have options affecting the whole game, such as sound volume.

Quit Button: This will close the game window.

Constraints: None

Feature: 2
Name: Timer

Description: The user will be able to see the time it took for race cars to complete a race.

Design Details: The timer will be positioned to the top-left of race tracks. The timer will start when a race starts, counting up from the number 0, and stop when all cars in the race complete a preset number of laps. For each car that finishes the race, their time will be recorded below the main timer. Their time will remain there until the user exits the race.

Constraints: The time counter must reside in the virtual machine; otherwise, users with faster computers will unfairly have faster times than users with slower computers.

Feature: 3
Name: Car Motion Code

Description: The user will be able to type in code that determines how a car will move (given a set of possible commands), and a virtual machine will read this code. The code will take effect in the next race entered.

Design Details: The tracks will be broken up into sectors, the number depending on the track (initial implementation may keep to four sectors each track). Code for each sector must be entered.

Commands:

Sector(x);

Where x is from 1 to the total number of sectors, this command must precede the following ones.

Speed(x);

Where x is from 0 up to a specified speed limit. The car will stay at this speed for the entirety of the sector.

MoveToPosition(position, section);

Where position is 'left', 'middle', or 'right' (of the road), and section is a number from 1 to the total specified number of subsections. This means that once the car reaches the subsection, it will immediately perform a movement to the left, middle, or right of the road.

Turn(direction, section);

Where direction is 'left' or 'right', and section again is a number from 1 to the total specified number of subsections. This means that once the car reaches the subsection, it will immediately attempt a turn to the left or right.

**The same subsection for the same sector cannot be entered as a parameter more than once in a set of commands.

Constraints: The code entered by users will have to be completely error-free for the VM to understand it. If it is not, the user will need to be prompted to carefully look over their code and fix it before re-entering it.

Feature: 4
Name: Mud Obstacle/Slow Obstacle

Description: Whenever a race car passes over mud or grass on a track, it will slow down.

Design Details: Each track may have no to some level of mud and grass. Mud appears visibly as brown tiles, and grass appears visibly as light green or dark green tiles. A car that is positioned over one of these tiles will have a slower speed (a slower velocity in the direction it is traveling) than when it is positioned over the normal road tiles.

Constraints: None

Feature: 6
Name: Multiple Track Selection

Description: The user will be able to choose from multiple premade tracks.

Design Details: Premade tracks will be located in the menu accessed through clicking the main menu's track button. Each of these tracks will have the same pixel style and at the minimum contain wall objects. Once a user selects one of these tracks, the selection will be clearly indicated by an outline or some other marker around or next to the track. To enter the track, the user will have to go back to the main menu and then press the play button.

Constraints: When more tracks are added to the selection, accommodations will have to be made to fit them all on screen. This may be in the form of a separate menu or a horizontal list that the user can scroll through.

Feature: 7
Name: Lap System
Description: There will be a lap system for determining the completion of a race.

Design Details: Each track will have a set number of laps. Three laps will be the default number unless another number is decided on for a track. After starting a race, a lap counter will be incremented whenever a player touches back to the starting line. After the lap counter reaches the total specified number of laps, the user will not be able to control the car anymore, and the car will stop moving. Text will then appear signifying the ending of the race, and the user will be able to press a button to return to the main menu or replay the track.

Constraints: None

Feature: 8
Name: Improved Car Movement

Description: The user will be able to control a car that moves similar to a real car in a 2D environment.

Design Details: The car will be able to move forward, backward and gradually slow to a stop when the user is no longer providing input. It will be able to rotate clockwise to turn right and counterclockwise to turn left.

Constraints: None

The user will have to first enter code for car attributes.

Car attribute commands:

Acceleration(x);
Handling(x);
Weight(x);
etc.

The sum of these numbers will not be able to surpass a specified limit.

If they do not enter one or more of these, values will be set to defaults.