FINAL PROJECT  
Drag Race

CS223 Section 2 Fall 2016 Semester

John Martincic

December 6, 2016

**Time log**

|  |  |  |
| --- | --- | --- |
| Date | Time Spent | Quick Summary |
| November 26 | 2 Hours | Started mapping out project and making a plan |
| November 27 | 3 Hours | Started Code, worked on main, file checking, and RNG functions |
| December 2 | 3 Hours | Designed manual mode function flow |
| December 3 | 4 Hours | Started Manual mode functions |
| December 4 | 4 Hour | Finished manual mode and race functions |
| December 5 | 6 Hours | Created display, results, print results, and organize racers functions. |
| December 6 | 8 Hours | Created Auto mode function, code Cleanup, Bug Fix, Documentation |
| Total Time | 30 Hours |

**Requirements**

1) Emulate a car racing game

2) Read a document containing the cars

2a) Has: Name, type, number, and color

3) Allow user to pick which car

3a) Cars cannot be same number

4) Use RNG for movement

4a) Cannot be more than 10 spaces

4b) Track max length is 90 spaces

5) User Inputs

5a) car.txt

5b) Auto/Manual Race

5c) stop or do another race

6) Auto mode

6a) Runs with 4 cars

6b) Displays locations in relation to other racers

7) Manual Mode

7a) 2 Players pick cars

7b) same as auto mode

7c) Except user provides input for next "roll"

8) Record to file

8a) Name of file --> Player raceresults.txt

8b) Name of driver and position

**Design of Functions Description of Functions**

Opens the project, declares requirements

fileCheck()

Main Function

Prototypes / Variable

Declarations

Include / Define

Functions

Introduction

Comments

Implements the needed library files and sets global variables

The set up for the modularizing of the file

The main function being run that calls all the others

Verifies the cars.txt file is there

manualRace() / autoRace()

Runs depending on user selection. Sets variables to be used for remaining function and calls them

race()

Runs the race with prior configured variables calls the final functions

organizeRacers() /

display

Sets all the racers positions to be zero to simulate start line

Will sort the results and display on screen, as well as writing them to a file.

results() / printResults()

**Test Cases**

1. User tested several incorrect menu choices to verify the menu was operating as it should, as well as not allowing duplicate choices. (Requirements 3 & 5)
2. The program opened the file “cars.txt” with the choices moved around or swapped for different ones. The program recognized this and would adapt to display the new data in the proper form (Requirement 2)
3. The user ran the program in automode. The game automatically showed the places of the car, as well as which one made it to the line fist, then exported it all to a file. (Requirement 6 & 8)
4. The user ran it in manual mode, not being able to select two of the same car and then having to press enter every time the cars needed to advance. (Requirement 3, 5)
5. The movement as well as which “vehicle” moved were decided in a random function based off the time. (Requirement 4)

**Conclusion**

As a whole, I feel that CS223 has been very beneficial to me as I have very little programming knowledge before taking this class, but now I feel like I have a good understanding of the “C” Language, allowing me to expand into other programming languages. This class confirmed to me that I should stay in the computer science field, and with this basis that this class has provided me I should be able to learn many other programming languages without severe difficulty.