**Unit 5 - Activity 5**

**Drag Racer Simulation**

**Part I**

In this activity you will create a simulation of a drag racer trying to drive 1,000 meters as quickly as they can. We will make this simulation more realistic by accounting for the air resistance acting on the car.

1. Answer the following questions as you get started.
   1. When should the car start to ‘thrust’ forward?
   2. When should the parachute be deployed to slow the car down?
   3. When should the ‘thrust’ be stopped?
2. Consider the 3 sections of motion that might occur during a drag race. (Hint: consider the drag force on the front of the car to be separate from the force from the parachute on the car.)

|  |  |  |
| --- | --- | --- |
| Before the car starts to move | While the car is speeding up | While the car is slowing down |
| Force Diagram: | Force Diagram: | Force Diagram: |
| Determine an expression for the acceleration, based on the force diagram above: | Determine an expression for the acceleration, based on the force diagram above: | Determine an expression for the acceleration, based on the force diagram above: |

1. How would we want to program the acceleration function to determine how this car will move?
2. What factor(s) will affect the acceleration calculation? What inputs should this function take?
3. Would we need to use condition statements (i.e. if x > 1000), or would a single expression work for the entire motion?
4. Complete a Design Recipe for the find-a function and write the body of the function here. Make sure you get your Design Recipe checked by your teacher first.

For this part of the activity you will be the driver.

* To apply the ‘thrust’ to the car, you will operate the ‘left key’. Press it once to start the thrust. Press it again to end the thrust.
* To apply the parachute, press the ‘space bar’. You only need to press the space bar once to deploy the parachute

You have two goals:

* Cross the finish line in the shortest time possible.
* Stop the car as close to the finish line as possible.

You will be given a score determined by combining your time to the finish line and the distance traveled beyond the finish line before stopping. Your goal is to get the *LOWEST* score you can.

1. Open the simulation found here and try your luck: <https://tinyurl.com/y7noxdlr>
2. What strategies did you use to improve your score?
3. Do you believe you could improve your score more by practicing more or by automating the process?
   1. Were you to automate the process, on which aspect(s) of the process would you need to focus your efforts?
   2. What computer programming skill would you need to use to maximize your efficiency (and thereby generate the lowest score possible)?

**Part II**

Now we will program the computer to control the thrust and parachute to try to achieve the lowest possible score.

1. Think back to when you controlled the thrust of the car. Under what condition(s) did you decide to turn the thrust on or off?
   1. Turned thrust on:
   2. Turned thrust off:
2. Under what condition(s) did you decide to deploy the parachute?
3. Was there any overlap between when you wanted the thrust on and the parachute open?

This time we will write two separate functions, one for the force of air resistance acting on the car itself and one for the force the parachute exerts on the car. These functions will be called f-drag, and f-chute. We will also need to write a new find-a function.

1. Complete a design recipe for each of these three functions. Keep in mind that they are no longer controlled manually, so not all three forces will be applied to the car the whole time. Get these design recipes checked over by your teacher before moving on.
2. Open the simulation found here: <https://tinyurl.com/ybp9kpqv>. Add your functions, as well as the motion functions next-x, next-v. Run the simulation and see what score it gets. How does this score compare to when you were operating the car manually?