How does the big bad chair’s code work?

The route –

You write a text file and say whether it’s a right or left turn (R or L), the angle you want the chair to turn, the speed you want the turn to be (lowest 20, max 80) and the wait time between turns (seconds). No spaces in between just use commas.  
Example: R,90,40,5

Run through of how it’s is meant to work: Neurostimulation study - **TestController2.cs**

From there you can click one of the keys 1,2 or 3 –

* 1: Will start the thresholding where it executes the thresholding text file 6 times with a 30s wait time between each round.
* 2: Will start the experimental rounds where it executes the original route first and once that is complete, it takes a minute break and then starts 5 rounds of the pseudo randomised routes.
* 3. Will complete one iteration of the original route then stop.

Run through of how it’s is meant to work: Field of view study - **MarcStudy.cs**

From there you can click one of the keys 1 or 2 –

* 1: Will start the experimental rounds where it executes 5 rounds of the pseudo randomised route, total of 15mins.
* 2. Will complete one iteration of the original route then stop, total of 3mins.

**TestController.cs** is the experimental file, not used in any studies

Lightly press these keys once and fast because it can be very easy for the program to think you’ve pressed the keys multiple times and then it all goes off the rails.  
Sometimes the chair can be a pain in the ass and not want cooperate so you might need to restart unity sometimes or exit execution.

Other control keys –

* Left arrow: Turns the chair 90 degrees to the left at a speed of 40.
* Right arrow: Turns the chair 90 degrees to the right at a speed of 40.
* Up arrow: Gets the current angle.
* Down arrow: Recalibrates the chair so that its current position/angle is now the default/zero.
* 0: Will move the chair to position zero/default.

3 Set up functions –

* ReadText()

This takes a text file, splits it by “,” and “\n”. Returns it as a string array.

* Dict(string[] values)

Used for the pseudo randomised routes.  
Needs a string array parameter, use ReadText() to get this parameter in the correct format. This takes the array of text, puts each line of instructions into its own list and adds that into a dictionary with the instruction id as the key and the instruction list as the value.  
Returns a dictionary of ids and instructions to look up.

* RandomOrder()

Creates a list of random values between a range. Returns a list.

2 Turning functions –

* CalcRotatingRight(int angle)

Check whether the chair has finished turning to the right, then sets the chair’s position to 0 (this is needed to get the chair to stop). Returns a bool either true or false.

* CalcRotatingLeft(int angle)  
  Does the same as above but for turning left.

2 Driving functions –

* Drive()

If you just want to iterate through a route and nothing else then use this function.  
This uses the ReadText() function to get a string array to iterate over. It then loops over the array, checks whether the first element in the row is R or L and then executes the turn depending on which direction it is. WaitUntill() is used to ensure the program waits for the turn to finish and then uses Wait() to get the program to wait the desired seconds.

This keeps going until it gets to the end of the array.

* RandomDrive()

This uses Dict(ReadText()) to get a dictionary of ids and instructions to look up and RandomOrder() is used to get a list of random numbers.   
A foreach loop is used to iterate over the random list of numbers (ids) and then look up that id number in the dictionary to access whether its R or L. It then either adds or subtracts the angle it is planning to turn to ‘log’ and if the ‘log’ is over 180 or -180 then it switches the direction of the turn to the opposite direction. If the total angle of ‘log’ is within 180/-180 then it executes the turn in the same way as Drive().  
This keeps going until it gets to the end of the randomised number list.

2 Automation functions –

* Threshold()

Counter starts at 1 (for round 1), a while loop is used until the counter reaches 7 (for 6 thresholding rounds). In the loop it checks whether flag is false (originally is set to false) and if so it executes Drive() to run the thresholding route. WaitUntill() is used to wait until flag turns true and the route is complete then the Wait() is used to create the 30s break between rounds.

* RandomAuto()

This is the same as Threshold() except in the beginning a round of the original route is executed using Drive() and then it moves onto executing 5 rounds of the pseudo randomised routes, with a 60s break between each round.

You need to use Wait() and/or WaitUntill() or else the chair will just execute everything in one go.