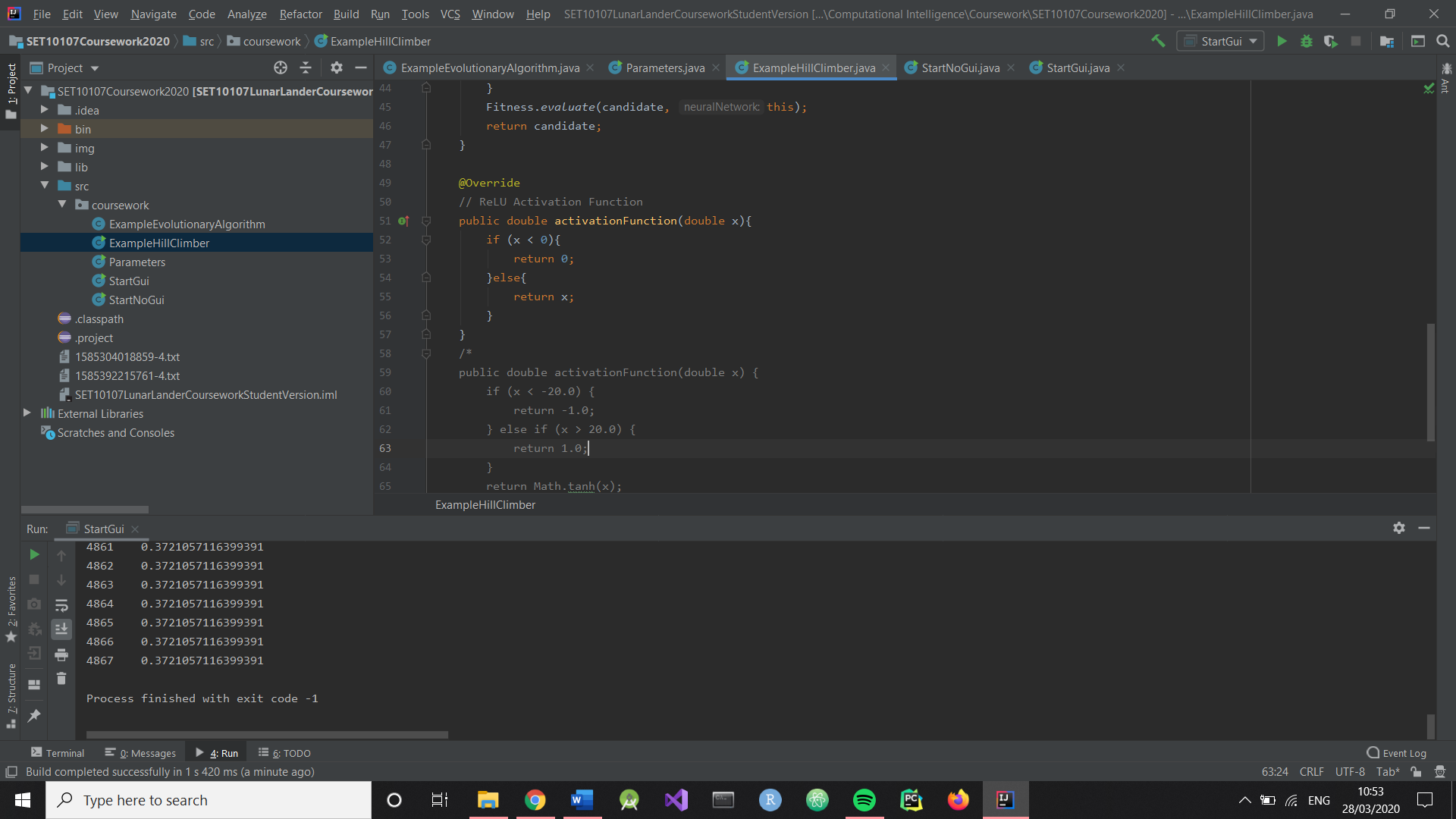
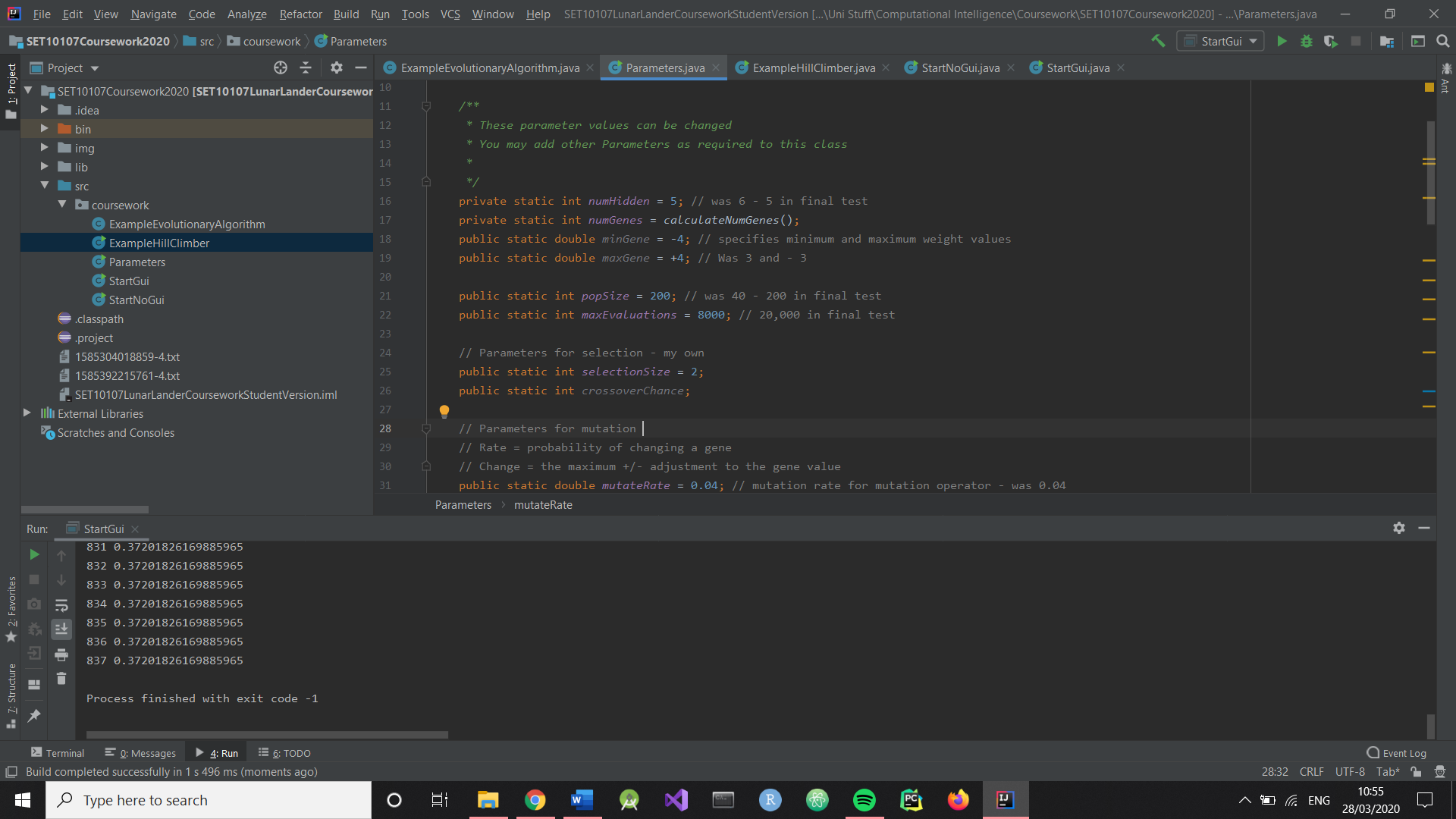
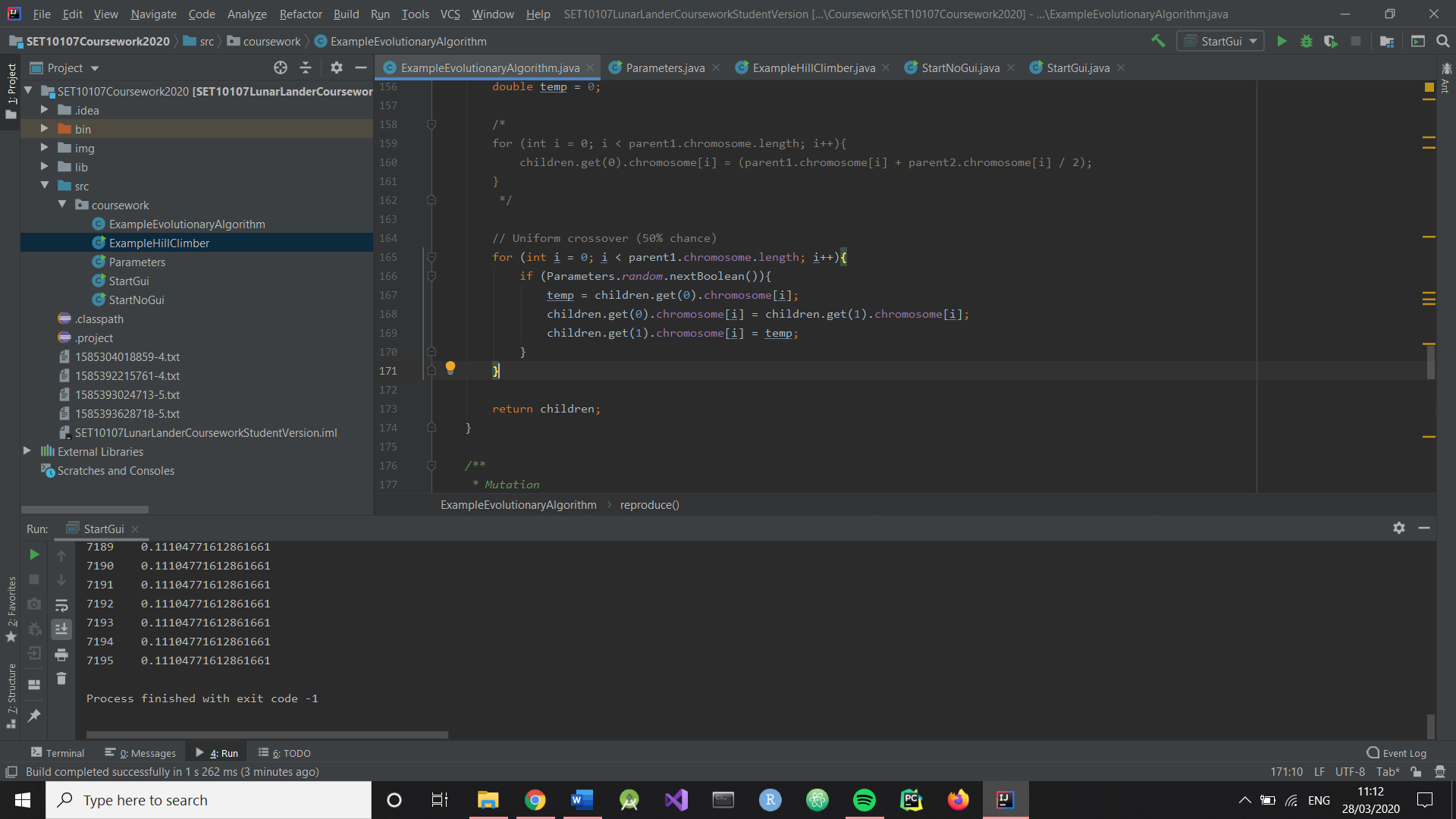
I did the testing and original project on a computer in D2 and haven’t managed to recover the code or the original report.

Started with the Arithmetic Crossover, however I found that the population was far too small, and I lost a lot of genetic material.

**Tried the ReLU Activation function – Didn’t work. Fitness flatlined**



**Set My Parameters to the Same as the Non-GUI Test (Except max evaluations)**

**Uniform Crossover with 50% chance – however it fails to help converge fitnesses**

**Implemented A single-gene crossover, but it didn’t change enough.**

**Implemented one-point crossover. Now there isn’t enough selection pressure**

Increased minimum and maximum gene from -2.5 and 2.5 respectively to -3 and 3. Increased selection size from 8 to 14

They can get good results if the weights start somewhere in the middle, but sometimes can’t ever recover. Need to check mutation to add more genetic information

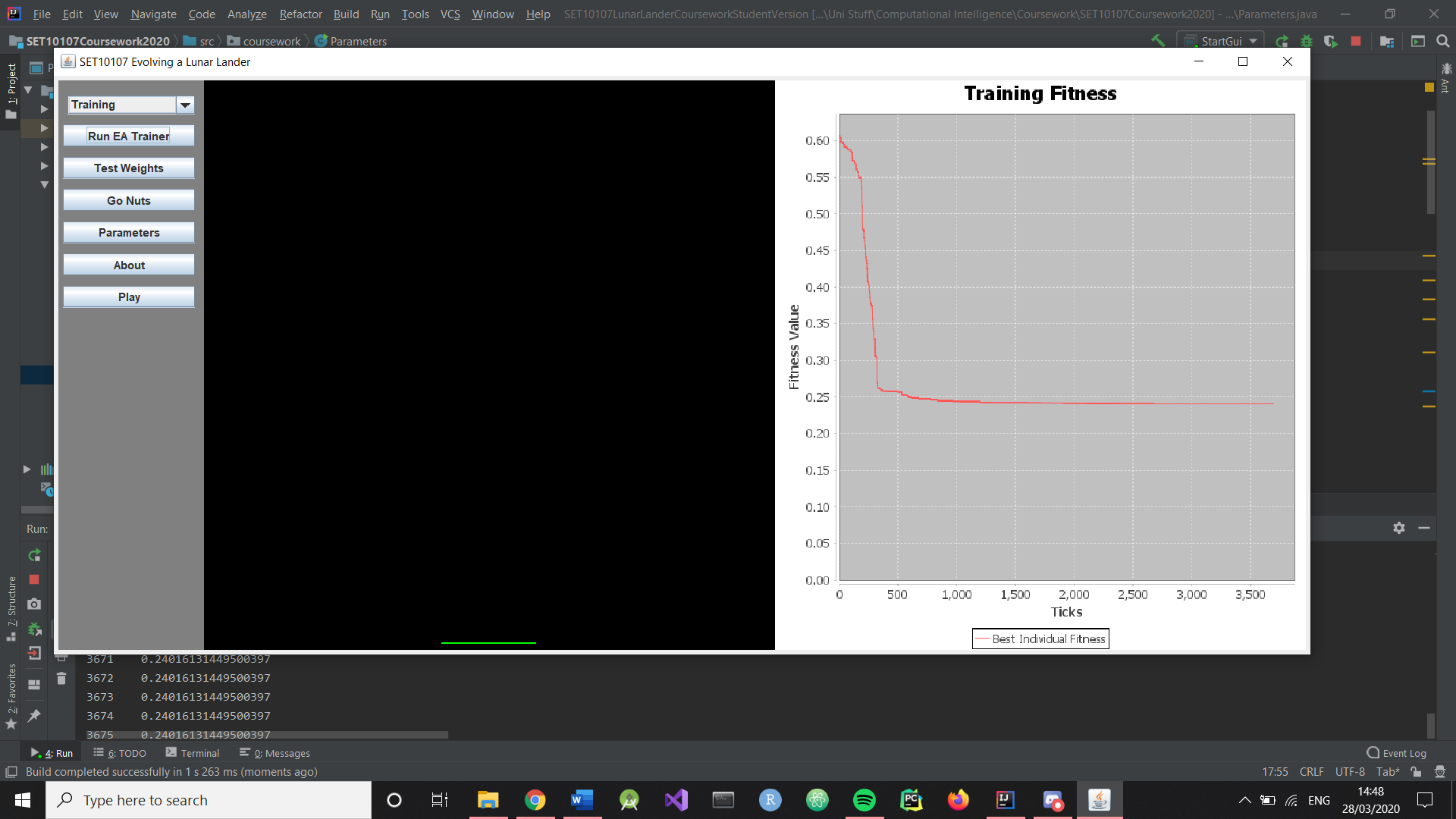
Upped mutation to .05 and .16 from .04 and .15

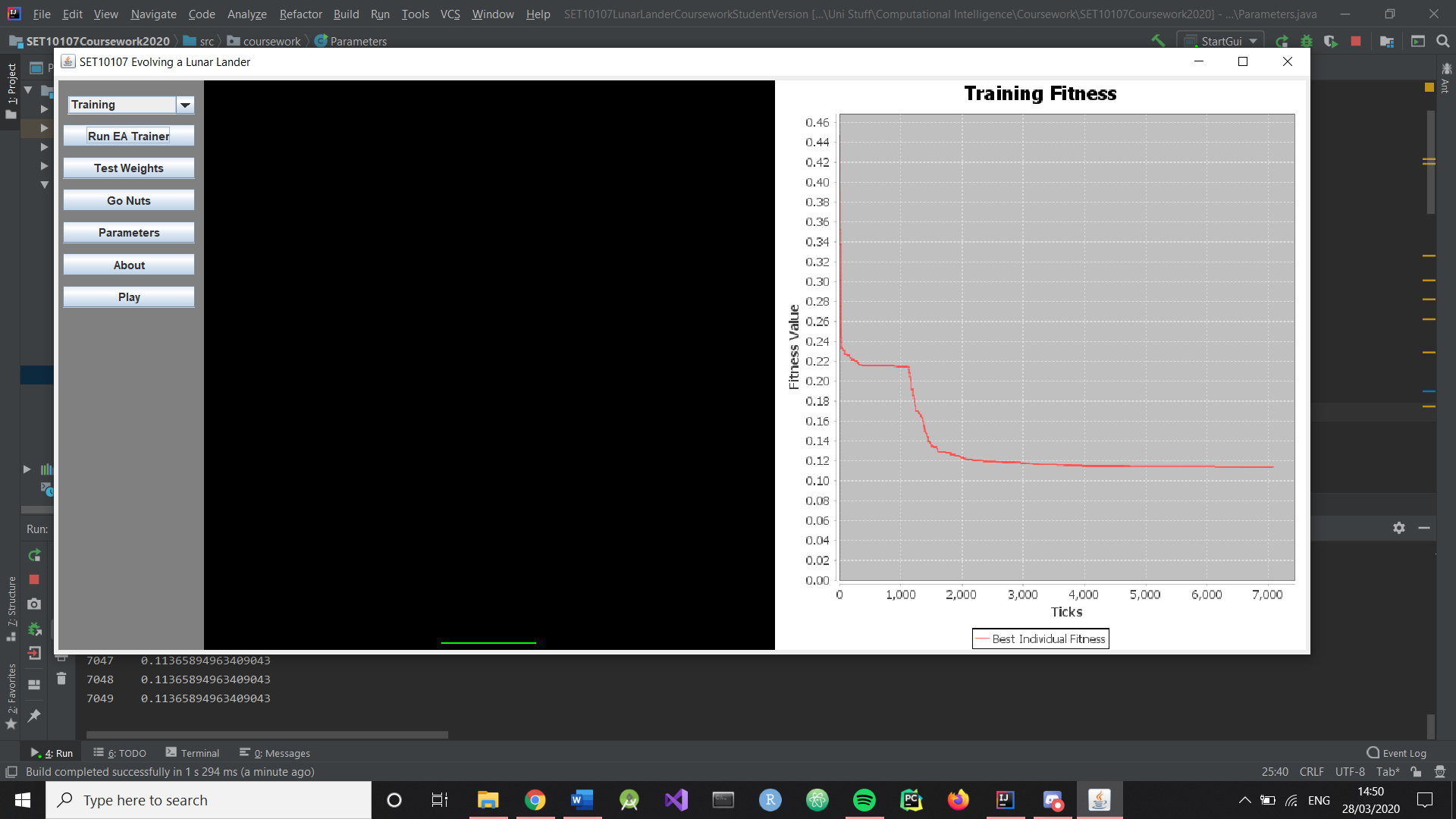
Anything more than 1-pt or 2-pt crossover seems excessive, as it will drastically alter the performance of the spaceship.

There was a bug in my selection causing it to return the first value in the array, not the best.

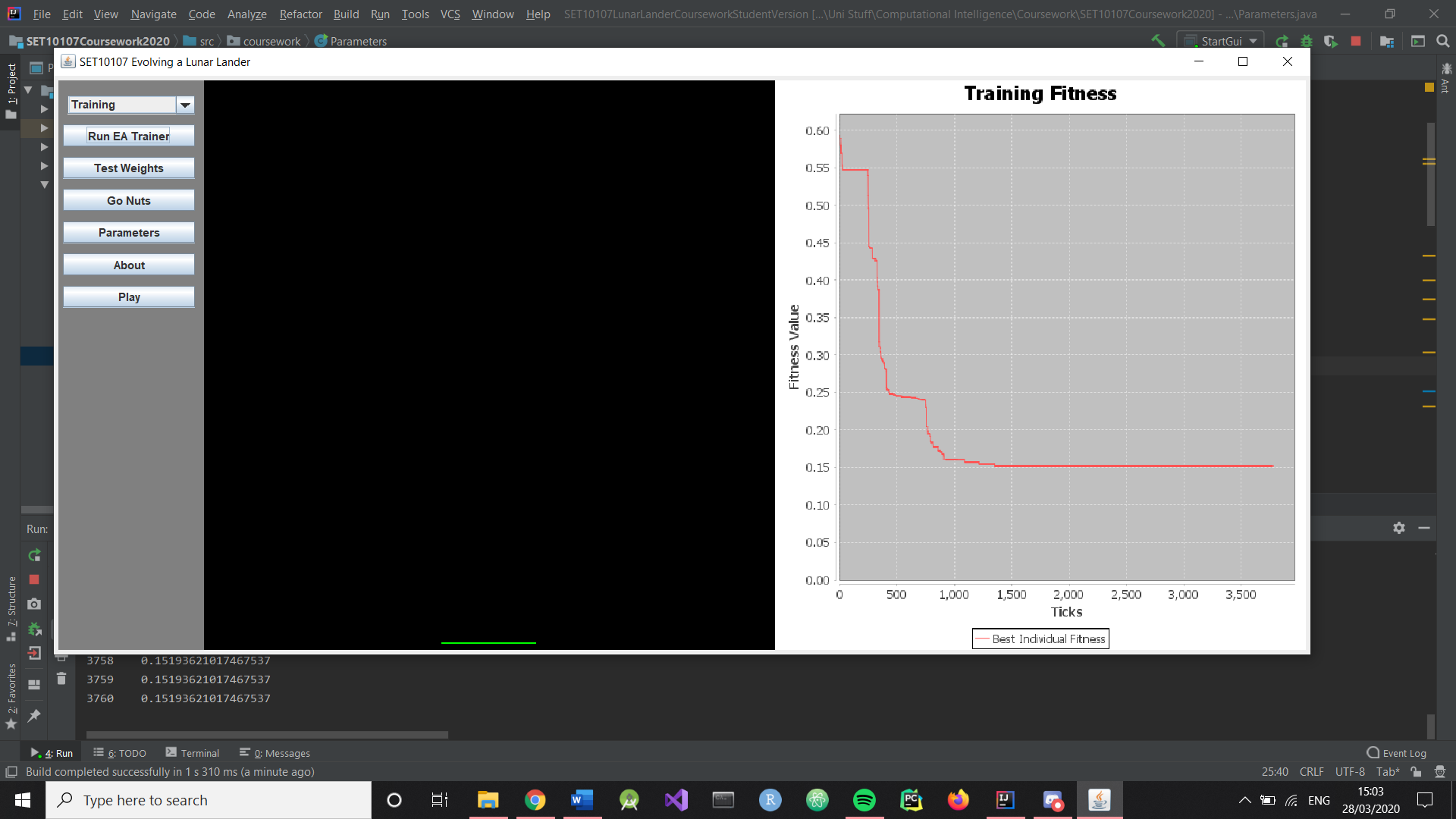
Increased selection pressure of the selection, and increased mutation chance (decreased rate).

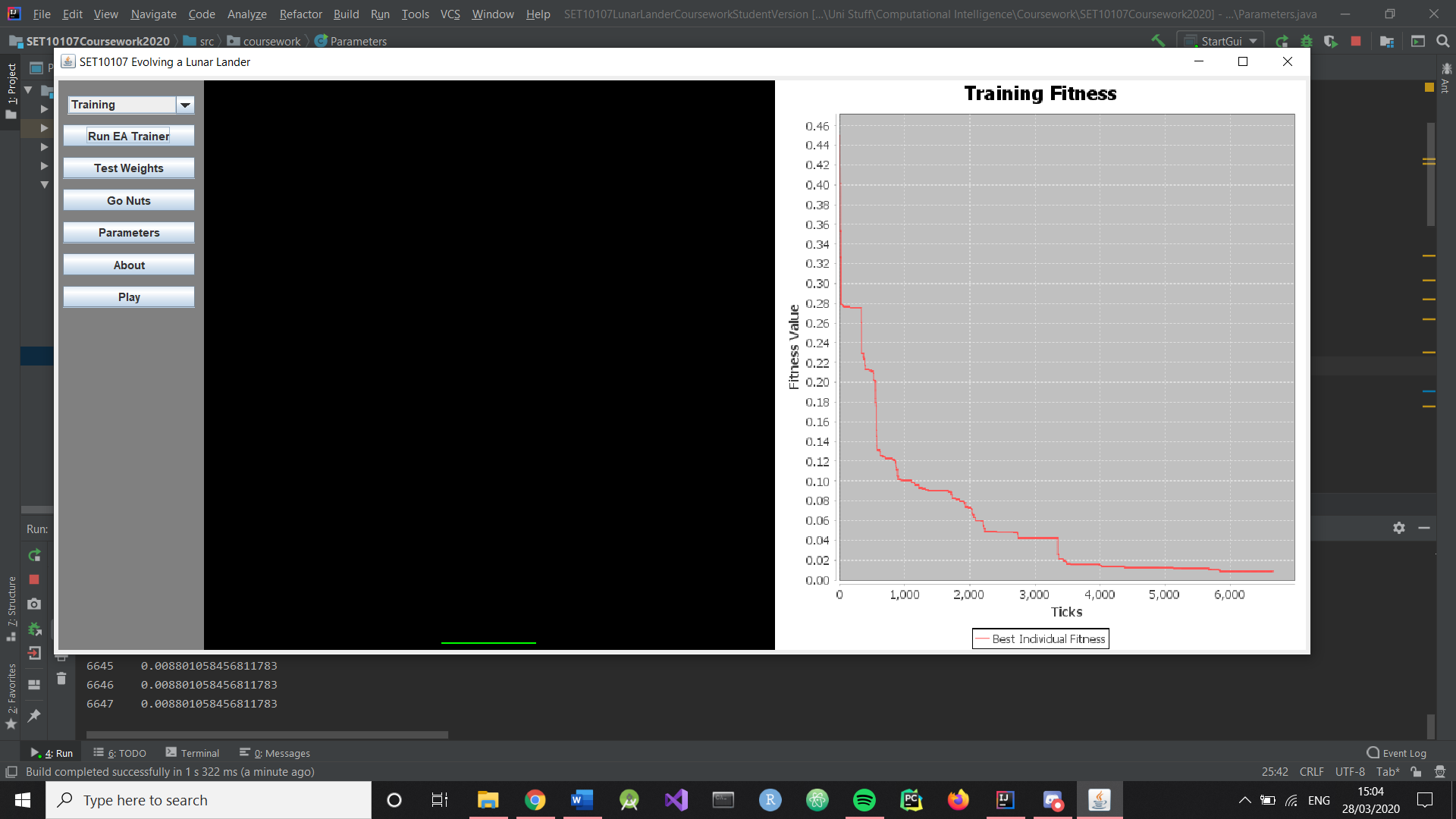
I changed the hyperbolic tangent function to the ReLU function, since it seemed far more appropriate for the task at hand. If > 0 then fire, otherwise don’t.

Found again that my selection pressure was too high. Keeps dropping all the way down and then stalling. Suggests that selection pressure is leading to premature convergence and a loss of genetic material.

In response, this was the graph from changing selection size from 30 to 10: 

Took minimum and maximum weights down for +2.5/-2.5 to +1/-1 since the ReLU function interacts differently with inputs. Performs even better with 0.5/-0.5 (average of 0.014). Note that some of the values actually mutated out of this range for that score.

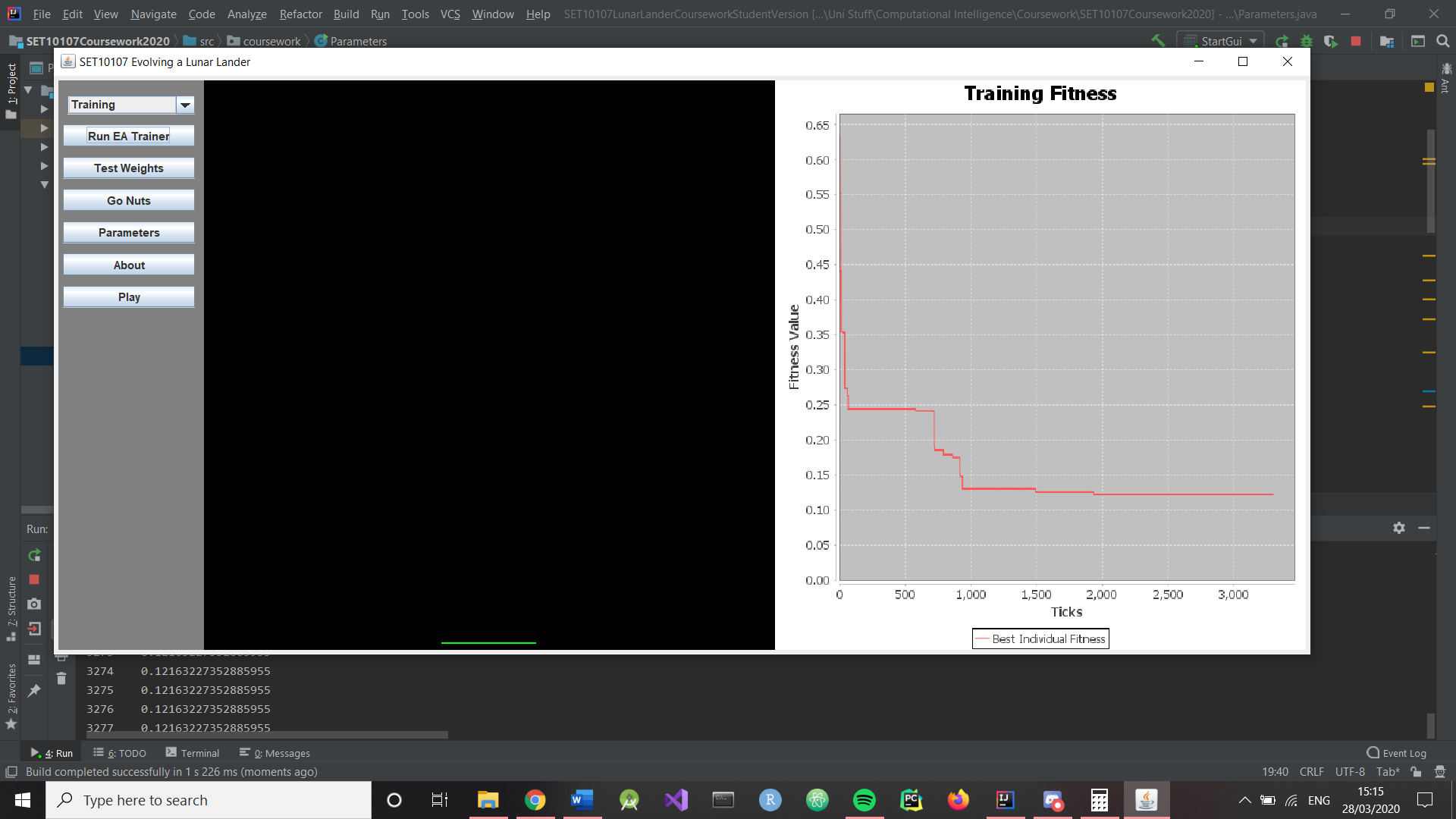
Low selection pressure (8)

Higher selection pressure (100) 

This selection pressure (100) meant that the optimal value had a 1/16 (6.25%) chance of being selected and compared against itself (1/4)\*(1/4), and the second had a 1/256 chance of being selected and compared against itself ((1/4)\*(1/4)) \* ((1/4)\*(1/4)).

At 120 that becomes 9/100 (9%). At 80 that becomes 4%. At 140 that becomes 12.25%.

Wrote method in the crossover class which selects a new parent if parent1 and parent2 are identical.

.1/.1 mutation, 100 selection size and .5/-.5 gene.

Consistently training under 0.02 fitness, even in cases where fitness plateaus previously: 