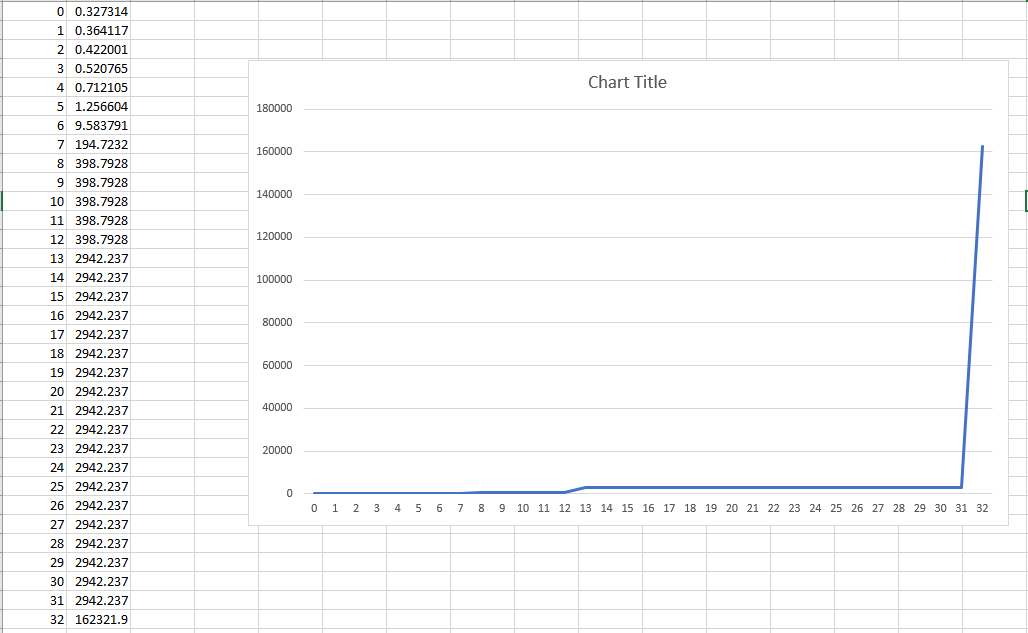
**C3 – Operators and Arrays**

3.1 – Part 1

|  |  |  |  |
| --- | --- | --- | --- |
| Run | RND\_INIT | Generations | Solution |
| 1 | 2 | 32 | 1.587402 |
| 2 | 3 | 14 | 1.587414 |
| 3 | 7 | 15 | 1.587402 |
| 4 | 5 | 9999 | 1.587426 |
| 5 | 4 | 9999 | 1.587747 |

In conclusion there are only certain values for the seed which yield good results.

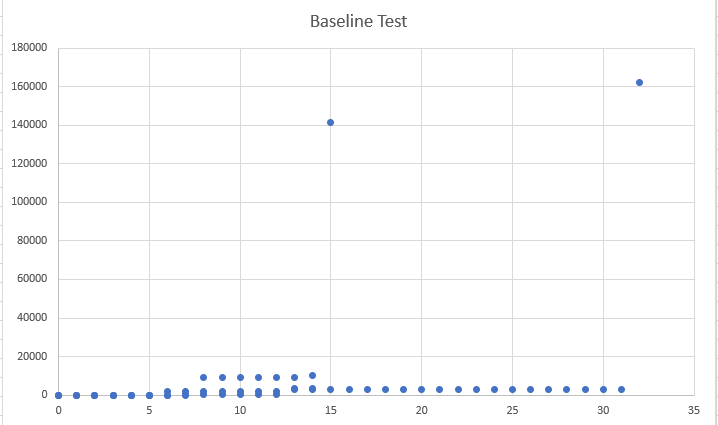
3.2 – Part 2



As the value of best\_ifit decreases (as it gets closer to the correct value), it tends towards 0. Therefore since the fit is 1/best\_ifit , the fit tends towards infinity. This produces an exponential graph.

3.3 – Part 3

I want to investigate how the population size affects the fit of the data. First I will start off with a baseline test. Baseline population is 100. I expect to find that as I increase the population, the program will find a close enough answer in less generations.



Now I will make a table of the different values of population size and see how it affects it.

|  |  |  |
| --- | --- | --- |
| Population Size | Final Fit Value | Graph |
| 250 | 16044.236328 |  |
| 500 | 15815.568359 |  |
| 1000 | 4205.834961 |  |
| 75 | 643.266907 |  |
| 300 | 25708.548828 |  |
| 350 | 5238.272949 |  |

Therefore, in conclusion, having a slightly higher population size makes the program finish in a fewer number of generations and also yields a more accurate result (a higher fit value). Although once the population is too high, it does not produce the wanted answer and just takes up the maximum number of generations. For the random seed of 2, the best population size which I calculated was 300 since it yielded a fit value of 25708.548828.

4 – Additional Work