1. I was very close to 50000 on my second generation, but I eclipsed 50000 by a lot by getting to 80000 on my third generation. I took about one and a half hours to run the 3 generations
2. This was my best strategy: (0.052678268691012775, -0.28662312907958354, -0.573205193062625, -0.21713816773625005, -0.24286785341578776, -0.996353193727862). The values measure the highest column height, deepest well depth, number of holes, number of lines cleared, the bumpiness of the board, and the sums of all the column heights, respectively. I think that the lower values were more impactful on the success of the tetris game. My program became the most effective when I added the sum of all the column heights to the heuristic, which is very close to -1 and is the lowest one in the value. It also became more effective when I added bumpiness, which is also a negative value, so it became more effective as I added more negative values. The only positive value in this strategy is also very considerably low. I did not expect this to occur and I thought the highest column height would have more of an effect, but it does not necessarily seem that way as shown by the results.
3. I actually did like this assignment even though I thought it was a bit frustrating. However, I think patience is an important aspect of coding and this assignment helped with managing and fixing an extremely large amount of code and also waiting for a long code to run. I think it should be kept.