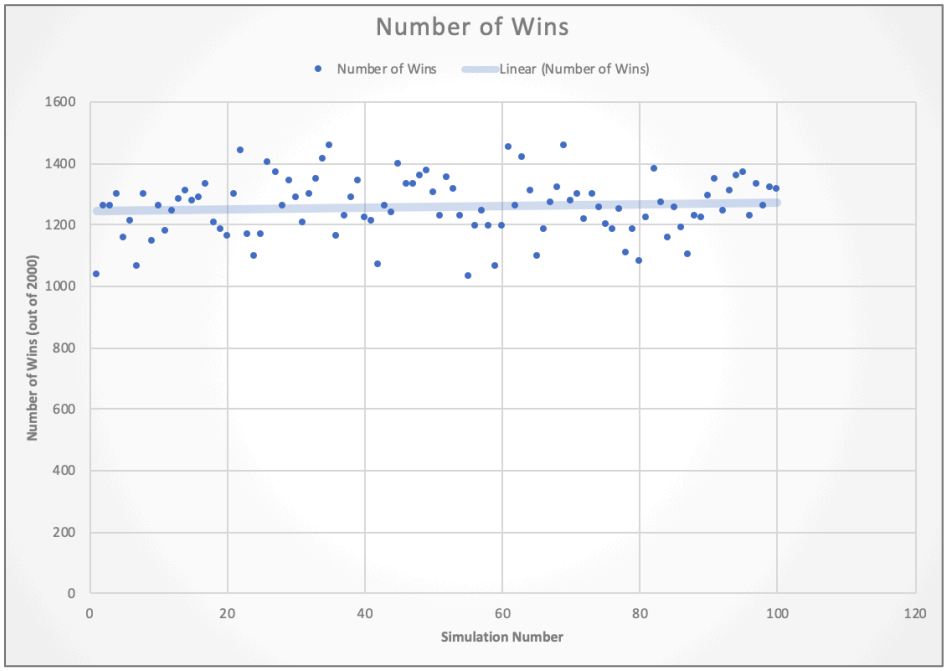
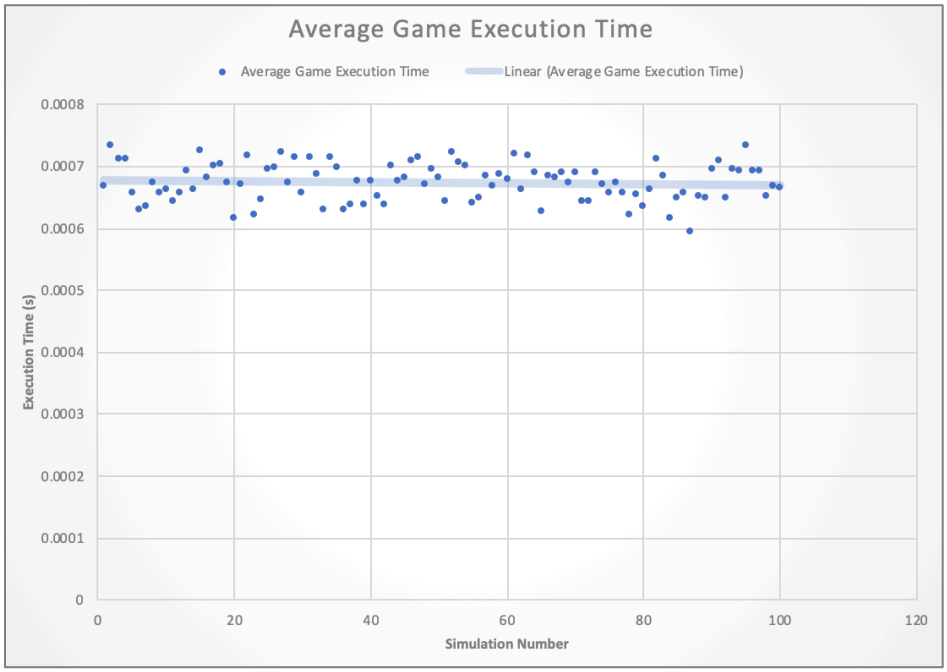
Solitaire Simulation Results

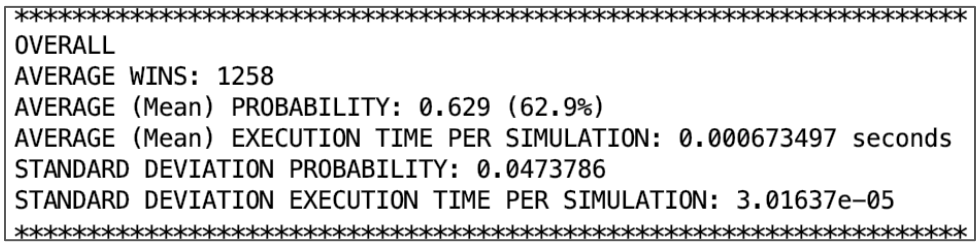
Our team created a Solitaire simulator. This simulation program plays a set number of solitaire games per simulation and runs a set number of simulations to determine the likelihood of winning a game. Our program is set to play 2000 games, and 100 simulations. For each simulation, the program outputs the number of games won (out of 2000), the likelihood of winning a game in the simulation, and the average execution time for each game in the simulation. Below is the data that we collected:





The two graphs above show the number of games won and the average execution time for each simulation round. It is easy to see that there aren't really any outliers in the data that we received. All of the values for the number of games won fell between 1020 and 1450, while all of the values for the average execution time fell between 0.00059 and 0.00074.

Solitaire Simulation Results



This figure uses all of the data we collected over the 100 simulation rounds and calculates mean averages for the likelihood of winning and the execution time per game. Here, we see that about 62.9% of games result in a win, and on average a game takes 0.000673497 seconds to complete. We also calculated the standard deviation for both the probability of winning each game and the execution time per simulation. We see that the probability of winning a game varies more than the execution time. Seeing this, we decided to run our simulation again. This time, we decided to do 200 simulation rounds, each with 5000 games. Below are the results of this: