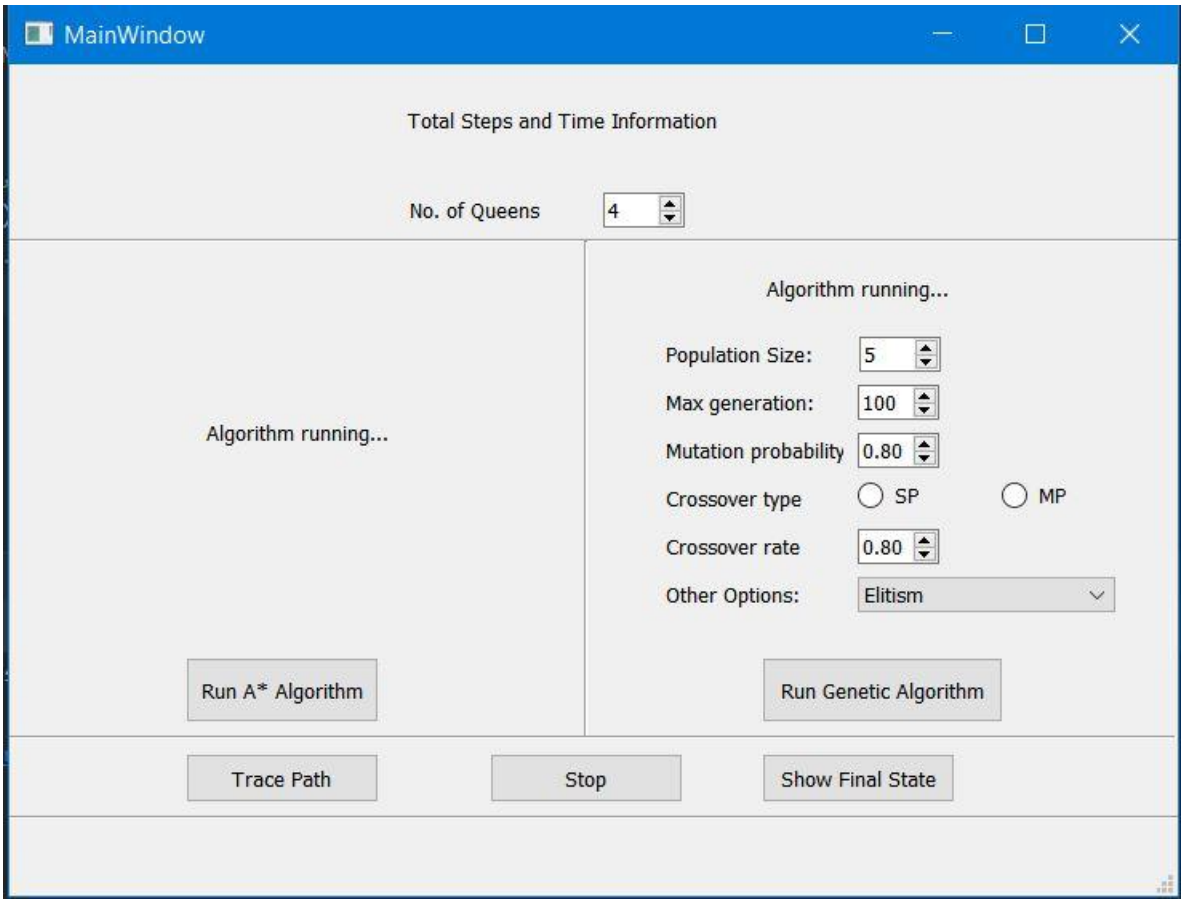
ICS 381 N Queens Programming Assignment

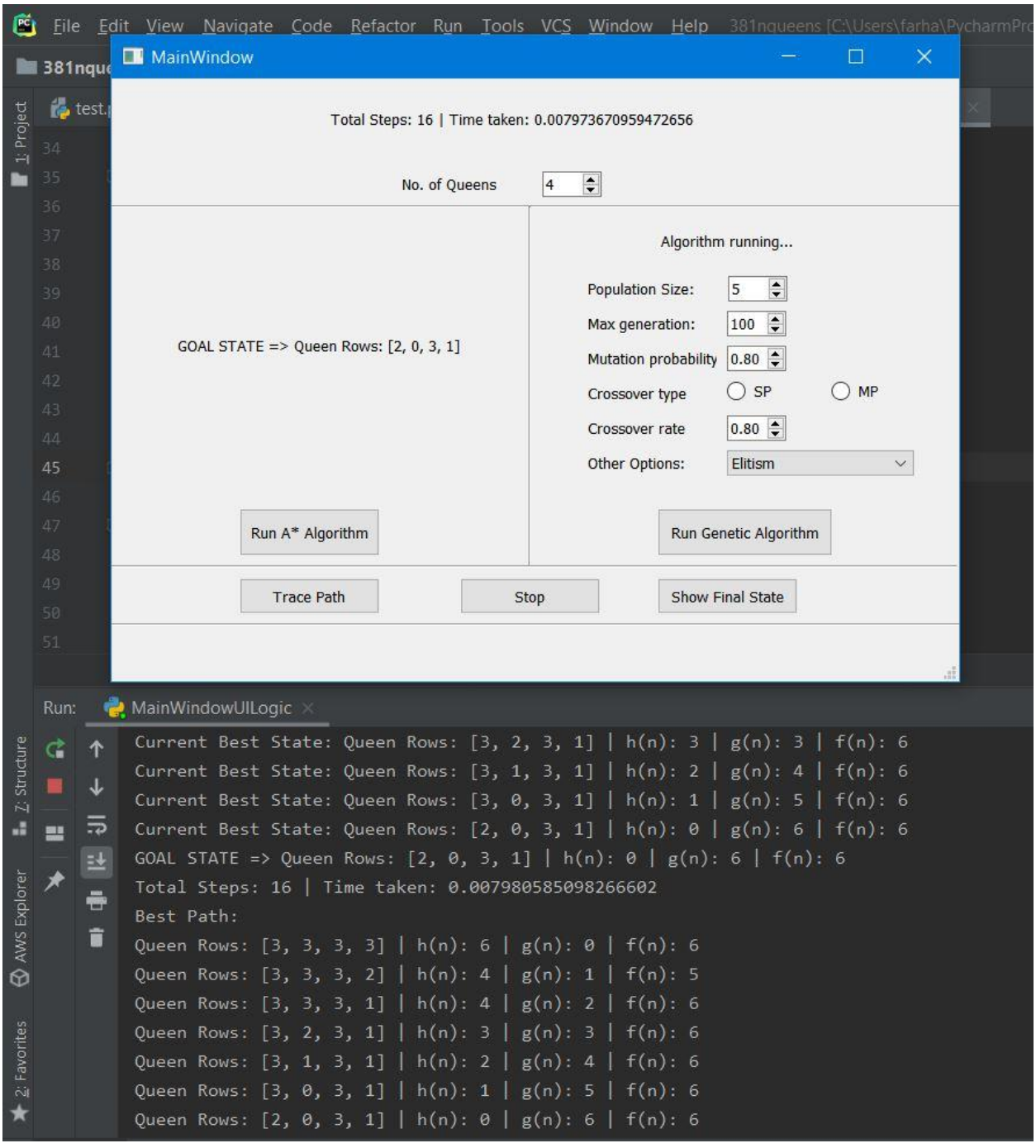
Farhan Mohammed Abdul Qadir 201771950

Sample Outputs

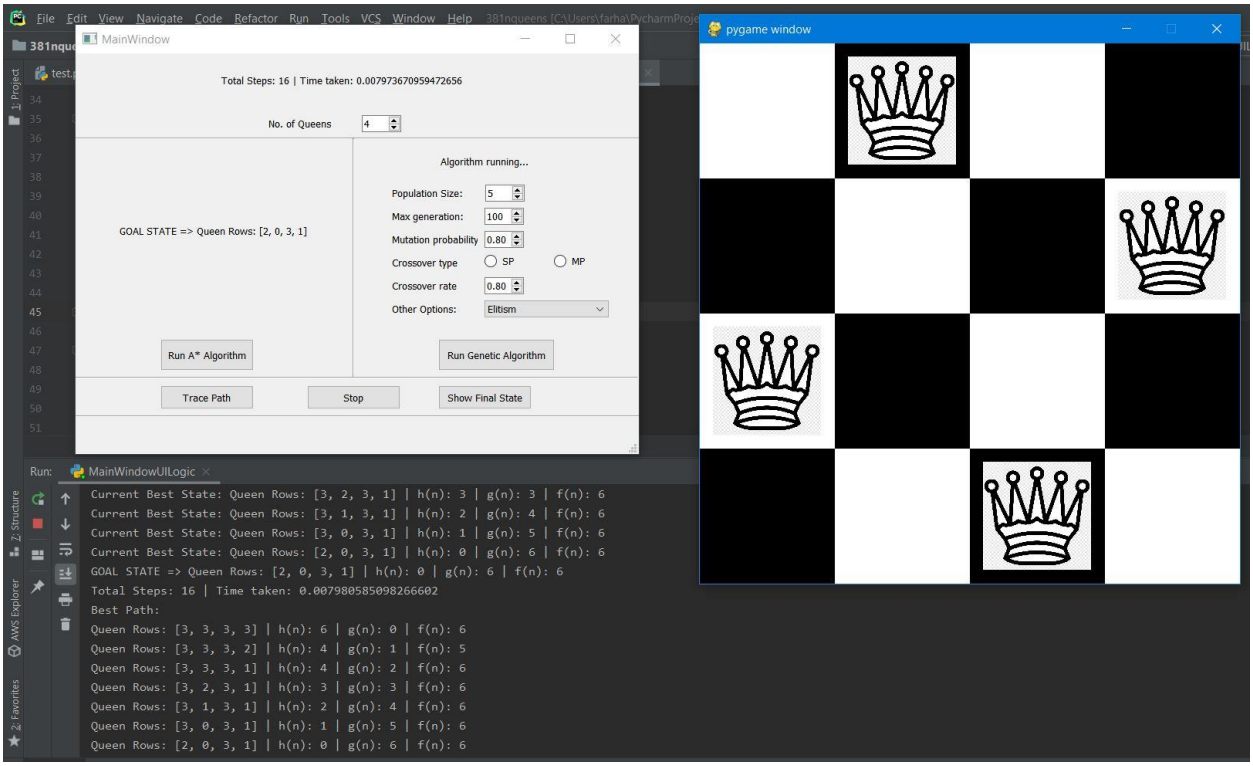
For both the algorithms, the parameters used for the sample outputs can be seen



GUI when program is run

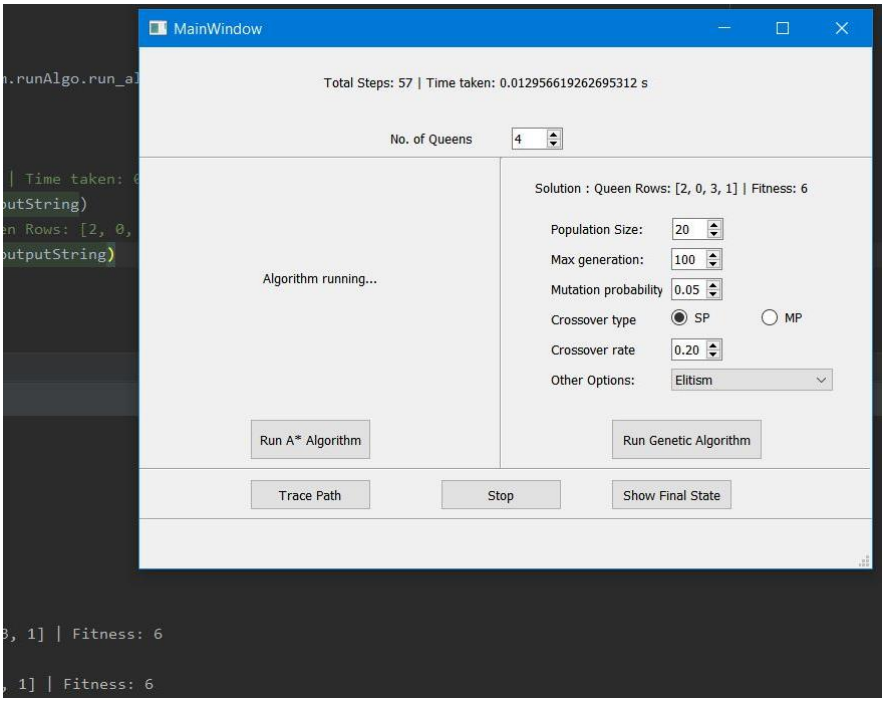


After clicking on run A* algorithm

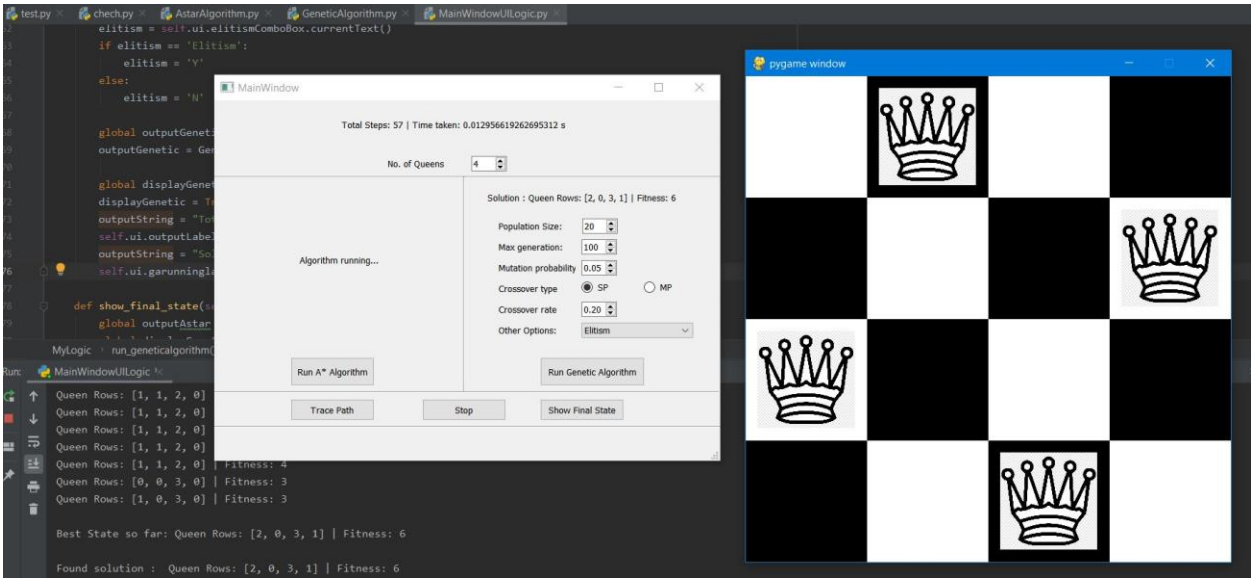


After clicking on show final state/trace path

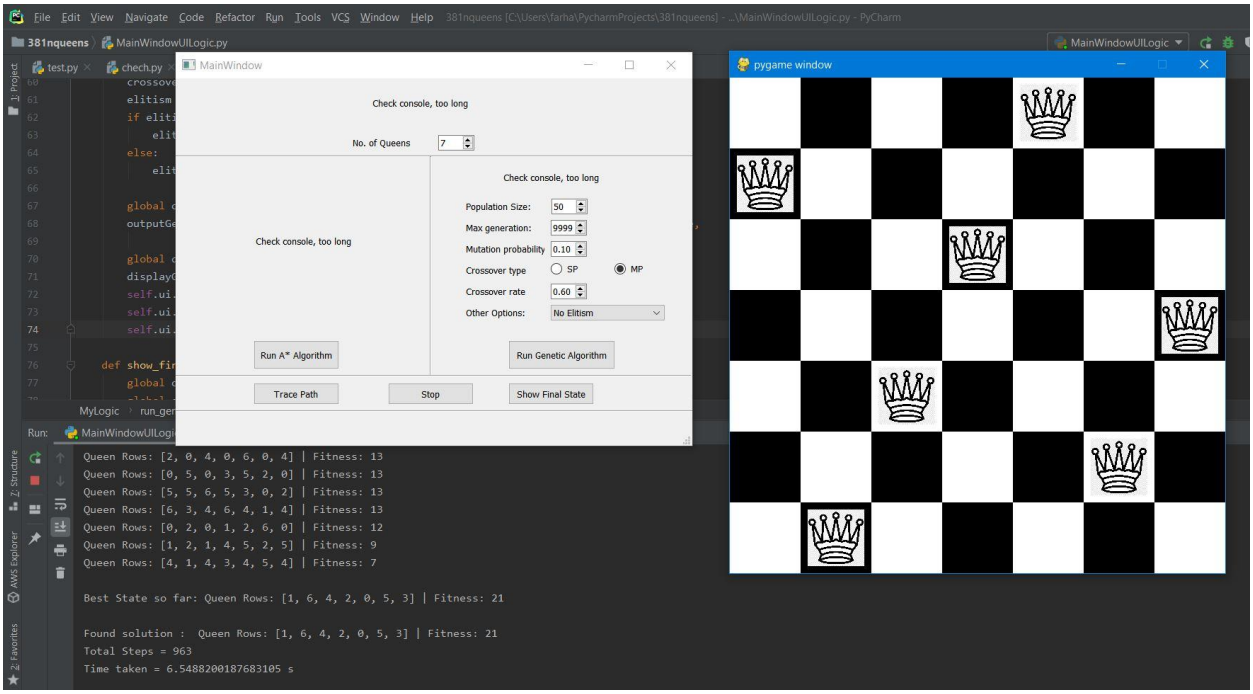
Note that trace path quickly displays the best path shown in the console one by one from the initial random state to the final goal state and so it tends to crash my laptop for a high number of steps as will be shown in the GUI



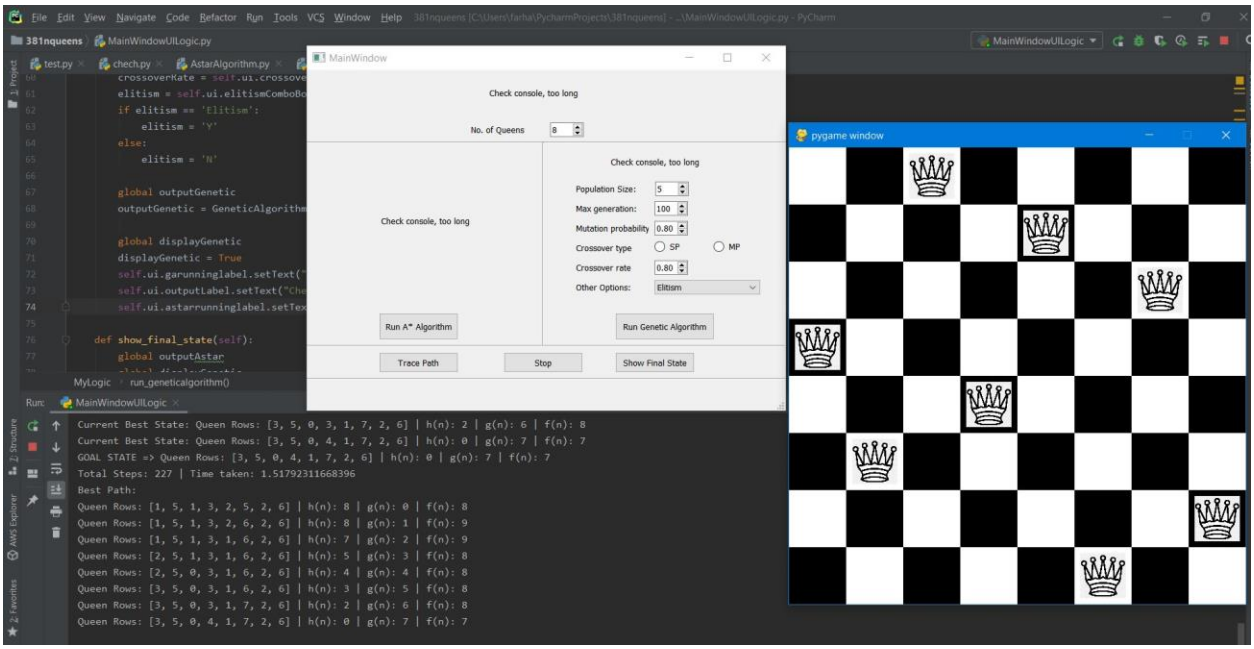
After clicking on run genetic algorithm



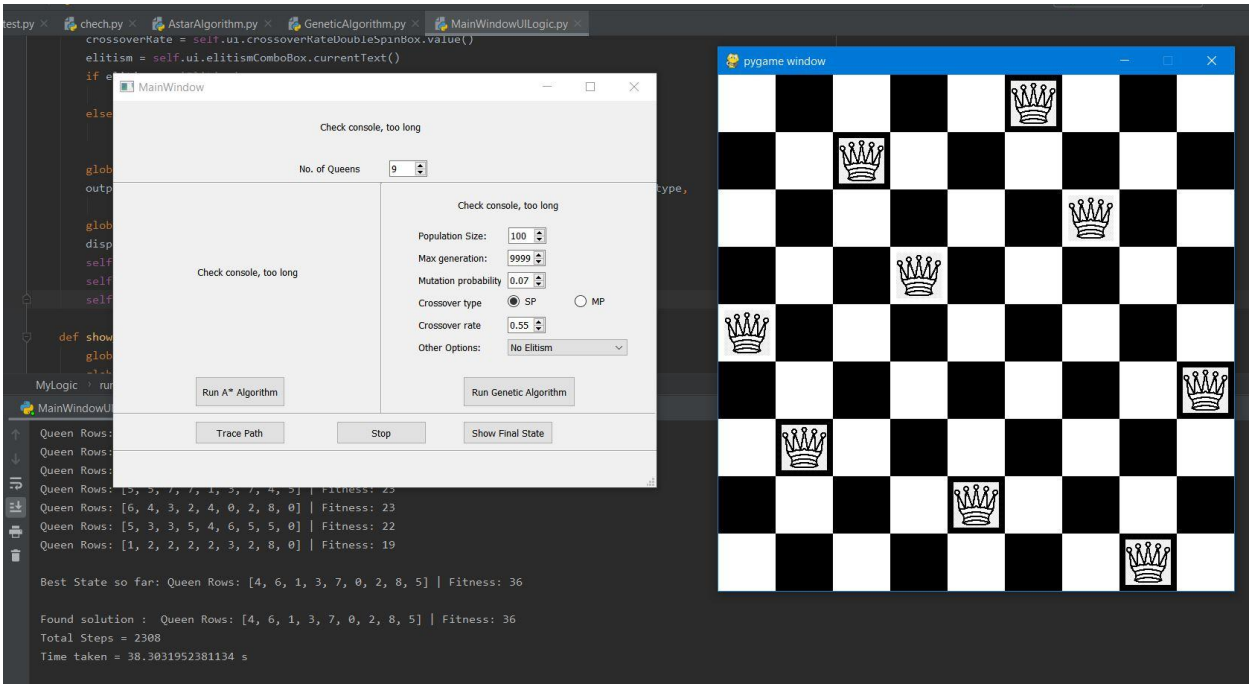
After clicking on show final state



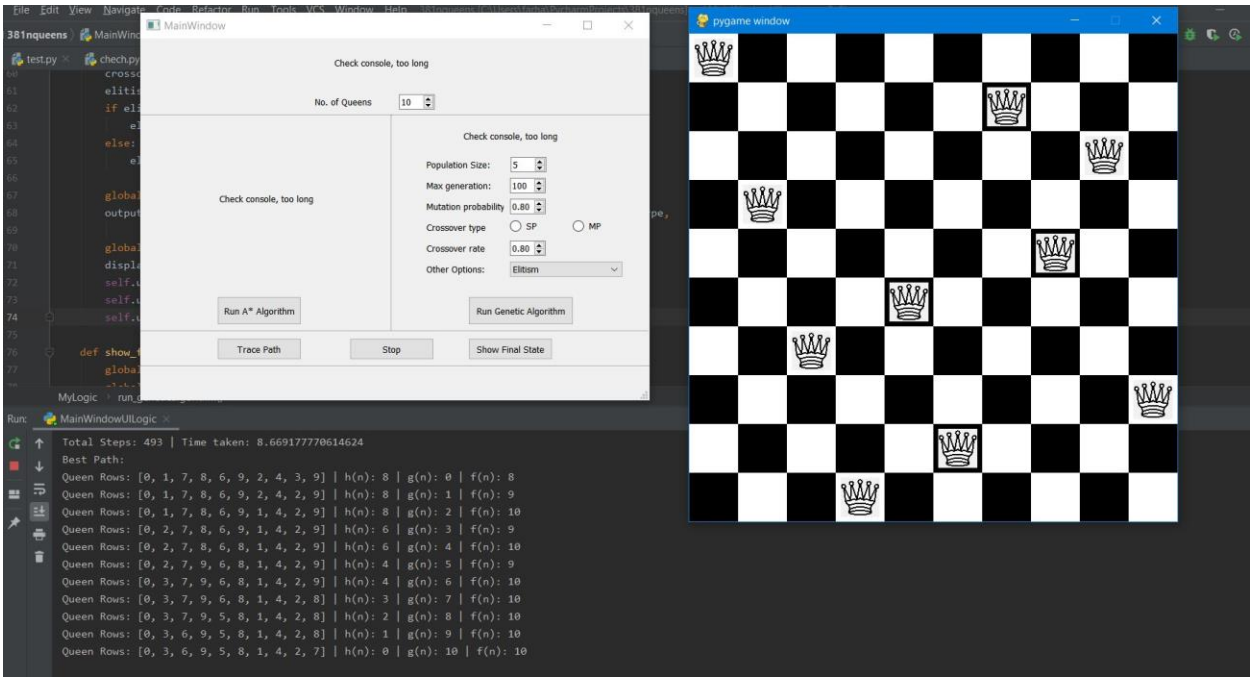
After clicking run genetic algorithm and show final state with shown parameters



After clicking run A* algorithm and show final state with shown parameters



After clicking run genetic algorithm and show final state with shown parameters



After clicking run A* algorithm and show final state with shown parameters

After this I ran the program multiple times and selected the approximate average to check the time taken for both algorithms in order to compare with a different number of queens and genetic parameters as shown in the above screenshot for 9 queens

4 queens

A* = 0.007 seconds

Genetic = 0.014 seconds

5 queens

A* = 0.027 seconds

Genetic = 0.041 seconds

6 queens

A* = 0.36 seconds

Genetic = 1.72 seconds

7 queens

A* = 1.22 seconds

Genetic = 2.96 seconds

8 queens

A* = 2.358 seconds

Genetic = 19.94 seconds

9 queens

A* = 18.71 seconds

Genetic = 84 seconds

A* seems to be more optimized for my implementations. Main issue at higher N values with A* was the memory constraints and for genetic, it seemed to be a local maxima for the fitness as it would remain close to the same value for long periods of time in different generations. However, there were many

outliers for the runs as the initial state for both populations is generated randomly and there is always a chance of reaching the goal state quicker given a more convenient initial random state/population.