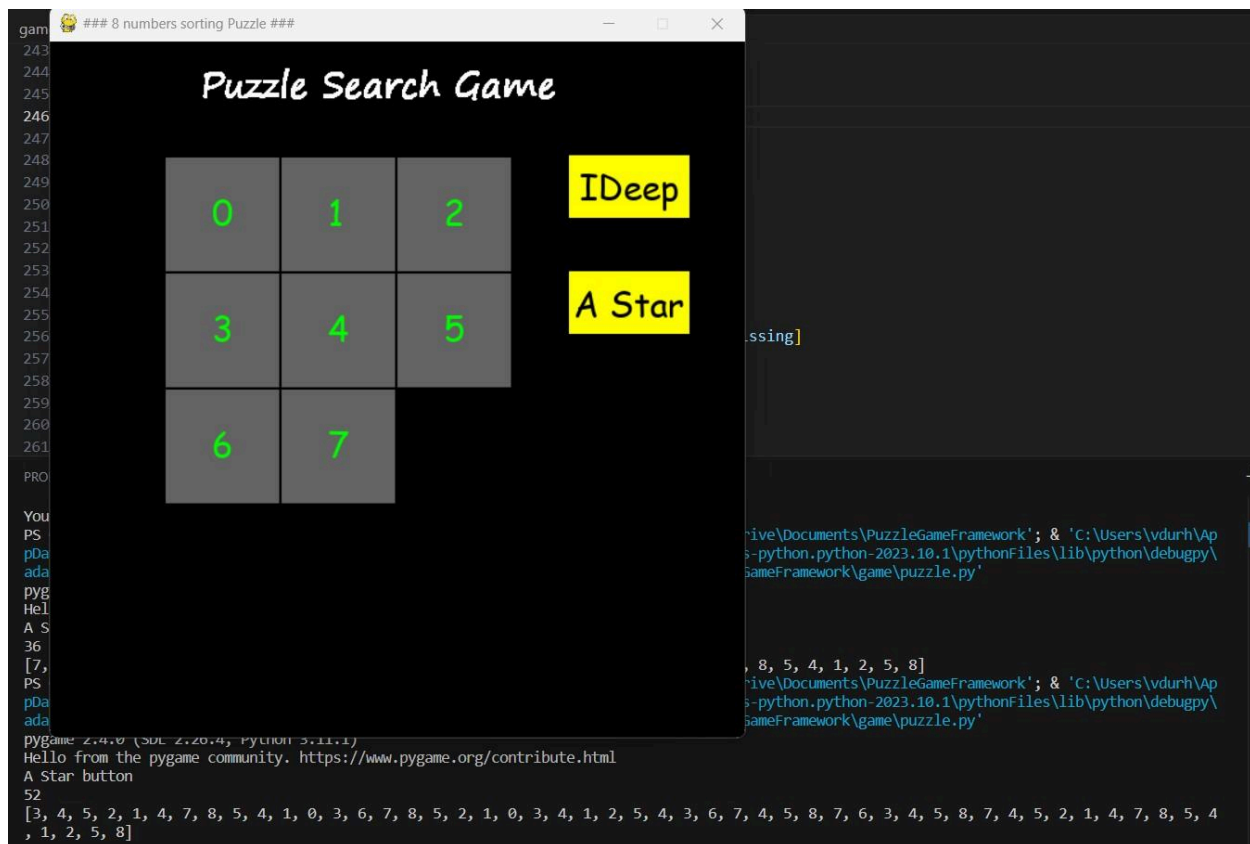
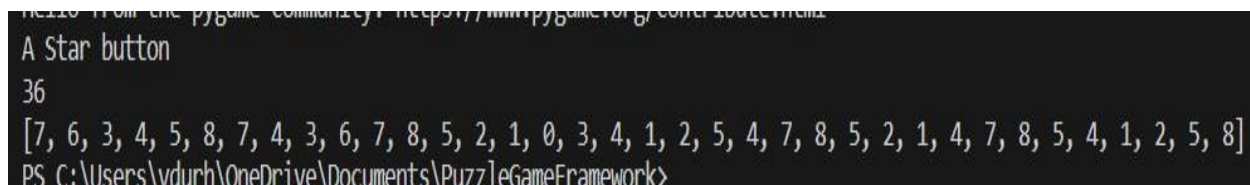


## A-Star Search

I was able to successfully implement the A-Star search on the 8 piece puzzle board. It was very similar to the Iterative deepening function, but I used the overall cost of each tile from its true winning position and used this to sort my nodes being found by, only expanding the puzzle with the lowest cost. I also fixed my coding in the iterative deepening part by expanding the functions within the main function, using a checkWinner function and a deepen function. My iterative deepening button can now solve puzzles up to a depth of over 15, but once you get to 20, it is unable to solve. The A\* search is easily able to solve a puzzle over 20 steps long to complete. The highest I was able to find was a 52 move win found by the A\* search.



- A\* search found a 52 move win on an extremely jumbled board



- Another impressive 36 move win found by A\* search