Notes (to be considered for report):

* Currently solving using recursion – if finding a solution takes long enough, it will cause StackOverflowError even if the puzzle is known to be solvable
* Could probably use some more diagnostic tools (e.g. total moves, time taken to find solution, etc.)
* DFS consistently causes StackOverflowError
* A\* has worse performance than BFS due to plateaus
  + f(n) for A\* never decreases as it goes deeper in the search tree
* How to know that h is admissible?
  + Run breadth-first search?
* Some puzzles are either unsolvable or take a very long time to solve (all three algorithms return StackOverflowError)
* Recursion good for testing, bad for accurate results (timeout before solution is found)
* Iterative implementation: if solution not found, essentially runs forever
* Only way to show “no solution found” message: Exhaust every possible state (takes a long time) or stack overflow