C++ report

A description of Design

I developed the iterative deepening A* algorithm as suggested and this is proven to find the optimal and was memory efficient.

I had a separate graph object that had functions for traversing and making relationships between adjacent nodes.

The main algorithms for finding the optimal path IDA* was sufficient for all 3 to 7 boards but it seemed to struggle so i used a greedy search which simply took the next best move rather than using iterative deepening.

Assumptions made

I assumed my machine or algorithm could not compute on the 8x8 and 9x9 scale so I used a different algorithm for that scale of board game. This may be because I simply didn't wait long enough for the algorithm to complete or I implemented the algorithm incorrectly. Either way I took an optimal path finding as considering time taken, which if toolong would defeat the purpose of finding the optimal path.

Instruction for compiling and linking program;

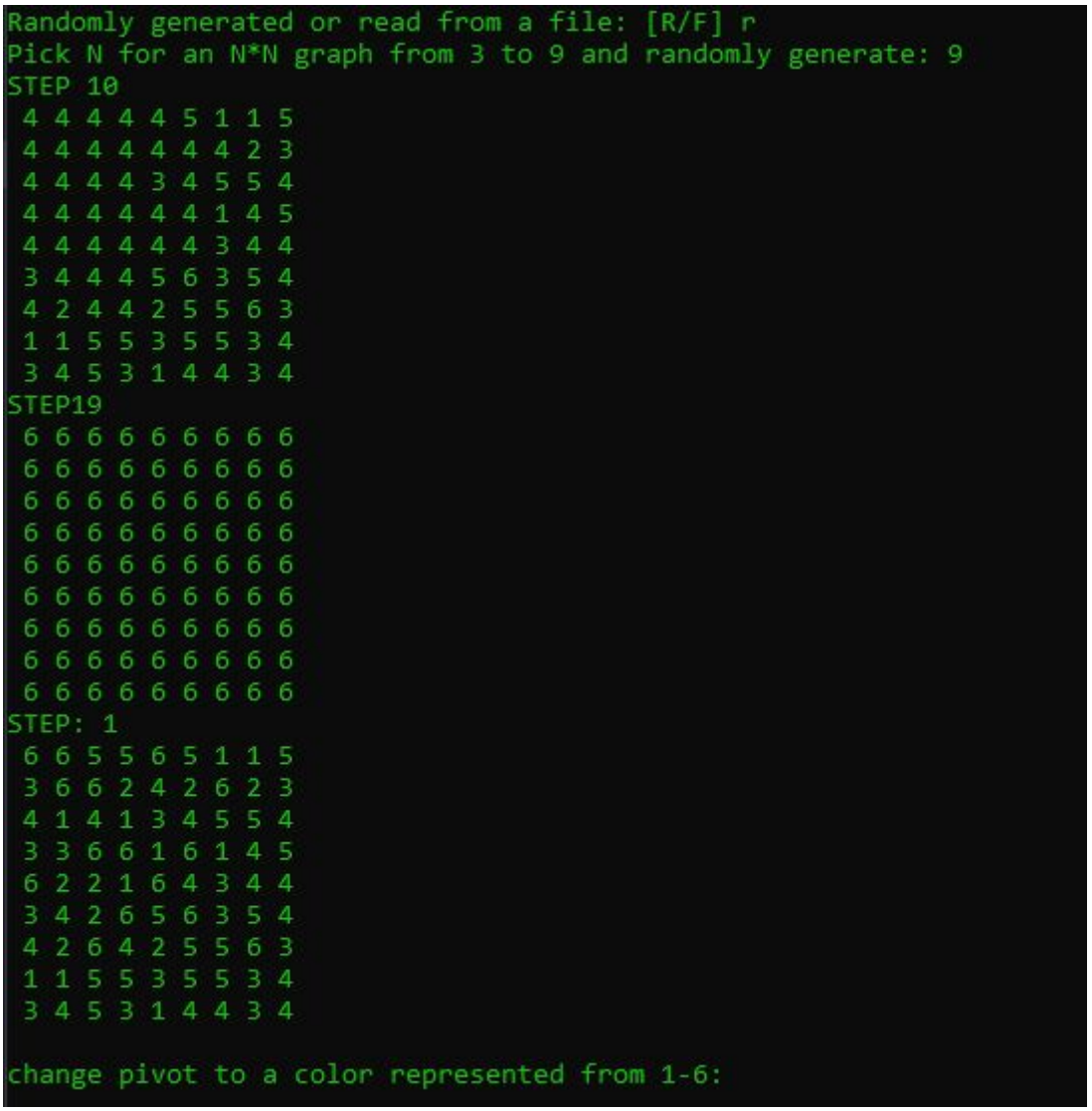
bash compileNlink

Generates Boardgame.exe

– A description of the main principles of the algorithm for searching the theoretical minimum number of steps;

The main principles of IDA* which was used for the majority of the project was to prioritise which state it would go to next and would not go to depths if it was unnecessary making it more memory efficient than the A*. It does this by working both in depth and breadth.

– An example output containing: the initial colours of a randomly generated board of 9 ×9 nodes, the colours of the board after the 10th step attempted by the algorithm that is designed to search for the minimum number of steps, and the final state of the board;



– If your program is not completely successful in producing the required outputs, then this report is the place to describe what is not working, why you think it is not

I believe my program does not work as intended for 8x8 and 9x9 as it does not find the true optimal as it took too long and I decided to use a different quicker and more reasonable solution timewise but the minimum is not found.