code instructions

part1

“blind_search” folder

1. breadth-first search
   a. `g++ bfs.cpp Node.cpp -o bfs`
   b. `./bfs`
2. depth-first search
   a. `g++ dfs.cpp Node.cpp -o dfs`
   b. `./dfs`
   The running process will fail. To make it work, the initial start state should be changed. An example state that can make dfs work is given in dfs.cpp.
3. iterative deepening search
   a. `g++ ids.cpp Node.cpp -o ids`
   b. `./ids`

“Astar” folder

4. A* search
   a. `g++ main.cpp Node.cpp -o astar`
   b. `./astar`
   The method implemented to find the next node to expand is not efficient enough so it may take a while to run the result.

part2

1. Build a training set

   `python construct_training_set.py`
2. train the neural networks models

   `python train.py`
3. test different A* search algorithms with 200 randomly generated 8-puzzles

   `python test.py`