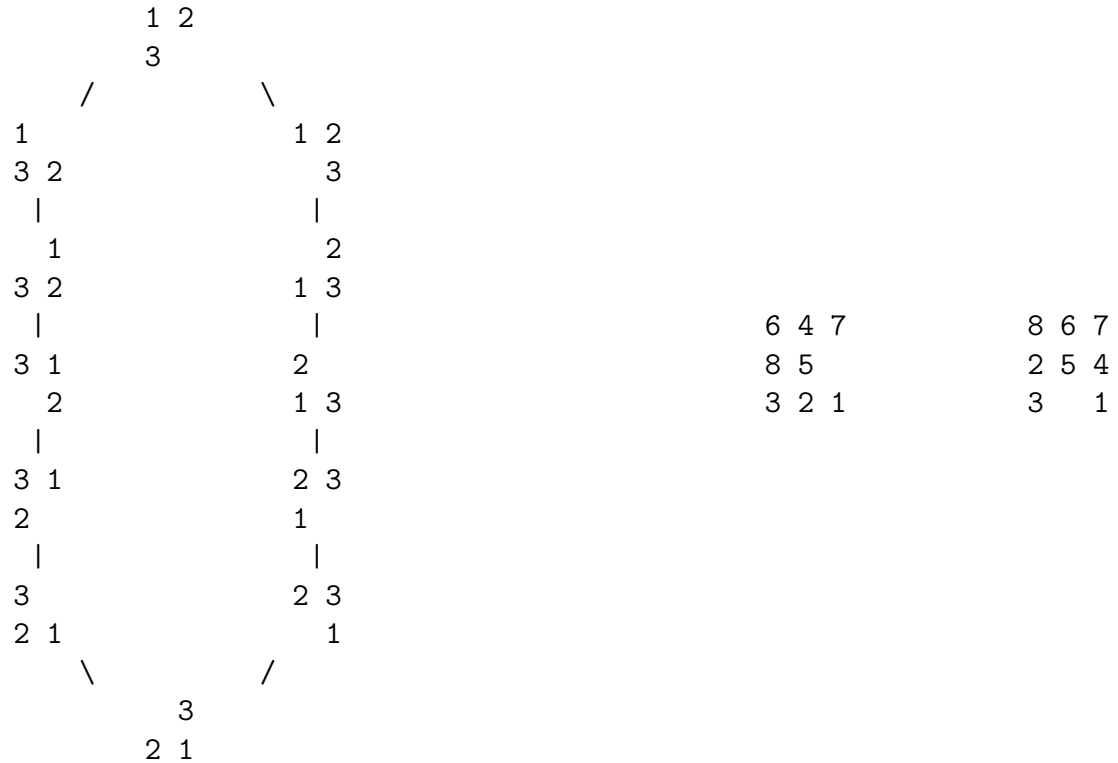**5.2.ii   On average, over all solvable 3x3 sliding tile puzzles, how many moves are needed to solve the puzzle? Explain briefly.** Some assigmnent questions are routine. This question was challenging. One way to answer this question is to run `stile_search.py` over all $9!/2$ solvable input states. Here is another way.

The 2x2 sliding tile adjacency graph below starts from the goal state and includes every solvable state: the number of moves in a best solution of a state is just its distance to the top goal state.

```
            1 2
            3
         /        \
       1             1 2
      3 2             3
       |              |
        1              2
      3 2             1 3
       |              |             6 4 7          8 6 7
      3 1            2              8 5            2 5 4
       2             1 3           3 2 1          3   1
       |              |
      3 1            2 3
      2             1
       |              |
      3             2 3
     2 1              1
        \          /
            3
          2 1
```

E.g. if we start from the goal state, a shortest solution makes 0 moves. If our start is either of the two states at the next level, a shortest solution makes 1 move, and so on. So, for the 2x2 puzzle, the average number of moves in a best solution will be $(0+1+1+2+2+3+3+4+4+5+5+6)$ divided by the number of solvable states, 12, so $(0 + 2 \times 1 + 2 \times 2 + 2 \times 3 + 2 \times 4 + 2 \times 5 + 6)/12$ $= 36/12 = 3$.

So, how do we find the number of nodes at each level in the corresponding diagram for 3x3? Take our 3x3 goal state, and change it by exchanging the values of the last two tiles, so (1 2 3 4 5 6 8 7 -). This gives us exactly the adjacency graph we want (except that we have exchanged the 8 and 7, which makes the state unsolvable and so guarantees that the algorithm will run through the complete state).

The output from `stile_search.py` on this unsolvable state shows us the number of nodes at each level of our diagram: 1 2 4 8 16 20 39 62 116 152 286 396 748 1024 1893 2512 4485 5638 9529 10878 16993 17110 23952 20224 24047 15578 14560 6274 3910 760 221 2.

So, over all 181440 solvable states, the average number of moves in a best 3x3 solution is $(0*1 + 1*2 + 2*4 + 3*8 + 4*16 + 5*20 + 6*39 + \ldots + 30*221 + 31*2)/181440 \approx 21.97$.

The data above shows that only two sliding tile puzzles — shown above right — have a 31-move shortest solution. How did I find them? Start from unsolvable 12345687. Modify `stile_search.py` to print out any puzzles at depth 31. Take these unsolvable positions and switch tiles 7,8.