1. In `simple/hex`, use `hex-3x3.py` to answer these questions:
   i) who wins, first player or second player?
   ii) how many nodes are searched when solving the empty board, x to play first?
   iii) how many nodes are searched when solving the empty board, o to play first?
   iv) why are the answers to ii) and iii) different?
   v) how do the answers to ii) iii) change if you uncomment line 107 and comment out line 108? explain.
   vi) why should 9! be an upper bound on the number of nodes that will be searched in solving 3x3 Hex from the empty board?

2. In `simple/hex`, use `hex-simple.py` to answer these questions:
   i) how many nodes are searched when solving the empty board, x to play first? How does that compare to your answer in question 1? Explain briefly.
   ii) Change the number of ROWS and COLUMNS to 4 and 4. Play x c2, o b4, x a4, and solve for o to play next. Who wins? How many nodes are searched?
   iii) Repeat this question for `hex-vc.py`. Why are the answers different than in ii)? (To answer this question, you will need to read functions `win_move` and `can_win` and compare them.)
   iv) Leave the number of ROWS and COLUMNS at 4 and 4. Play x c2, o b4, x a4, o b3 and solve for x to play next. What winning move did it find? Give another winning move for x here. Use the program to check your answer.
   v) From the position in iv), assume that x lets o play again, say o a3. Solve for x to play next. Who wins? Find all winning moves.
   vi) How does your answer to v) help you answer iv?

3. How could you modify `hex-vc.py` so that it solved 5x5 Hex positions a bit faster than it does now?