REVIEW EXERCISE 03

Question 1. The *N*-queens problem requires you to place *N* queens on an N × N chessboard such that no queen attacks another queen. (A queen attacks any piece in the same row or column or diagonal). Here are some important facts:

- The states are any configurations where **all** *N* queens are on the board, one per column.
- The **moveset** includes all possible states generated by moving a single queen to another square in the same column. The function to obtain these states is called the **successor** function.
- The heuristic function **h(state)** is the number of **attacking** pairs of queens.

a) Consider *N*=4. How many states are there in total? Explain your answer.

<hr>

<hr>

b) For each state, how many successor states are there in the moveset? Explain your answer

<hr>

<hr>

c) What value will the heuristic function **h(state)** return for state *S* shown aside? Explain your answer.

<hr>

d) Use some hill-climbing variant that can lead to a solution. Draw the search tree from S (Only draw the branches that lead to a solution; for each node on the tree, write down its **h( )** value).

Question 2. Consider the game search tree shown below, in which a square denotes MAX's move and a circle for MIN's move. Write down the utility value for each node of the tree using Minimax algorithm. Show where alpha-beta pruning occurs by crossing out corresponding branches. Briefly explain for each case.

≥3

≤3   ≤3

3 ≥3   ≥15   3

3   ≥3   ≥3   ≥3

2   1 5   2   ≤2

≤2

3	17	2	12	15	25	0	2	5	3	2	14