Assignment B-1

Aim:

To understand and solve 8-Queens Problem using backtracking.

Problem Statement:

8-Queens Matrix is Stored using JSON/XML having ïňĄrst Queen placed, use back-tracking to place remaining Queens to generate ïňĄnal 8-queen's Matrix using Python. Create a backtracking scenario and use HPC architecture (Preferably BBB) for computation of next placement of a queen.

Input:

8-Queens and empty playing board

Output:

8-Queens placed in non-conflicting position on playing board

Theory:

8-Queens Puzzle:

The eight queens puzzle is the problem of placing eight chess queens on an 8x8 chessboard so that no two queens threaten each other. Thus, a solution requires that no two queens share the same row, column, or diagonal. The eight queens puzzle is an example of the more general n-queens problem of placing n queens on an nxn chessboard, where solutions exist for all natural numbers n with the exception of n=2 and n=3.

Variations of n-Queens Problem:

8-Queens Problem:

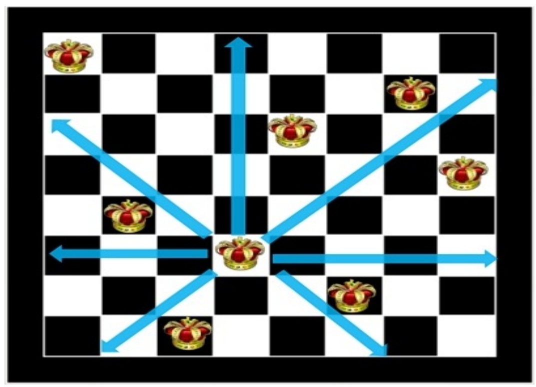

Figure 1: Example solution for 8-Queens problem

4-Queens Problem:

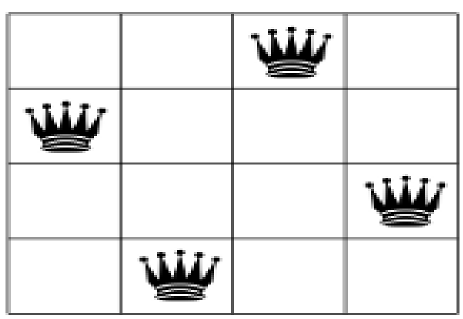

Figure 2: Example solution for 4-Queens problem

Backtracking Algorithm:

Backtracking is a general algorithm for finding all (or some) solutions to some computational problems, notably constraint satisfaction problems, that incrementally builds candidates to the solutions, and abandons each partial candidate c ("backtracks") as soon as it determines that c cannot possibly be completed to a valid solution.

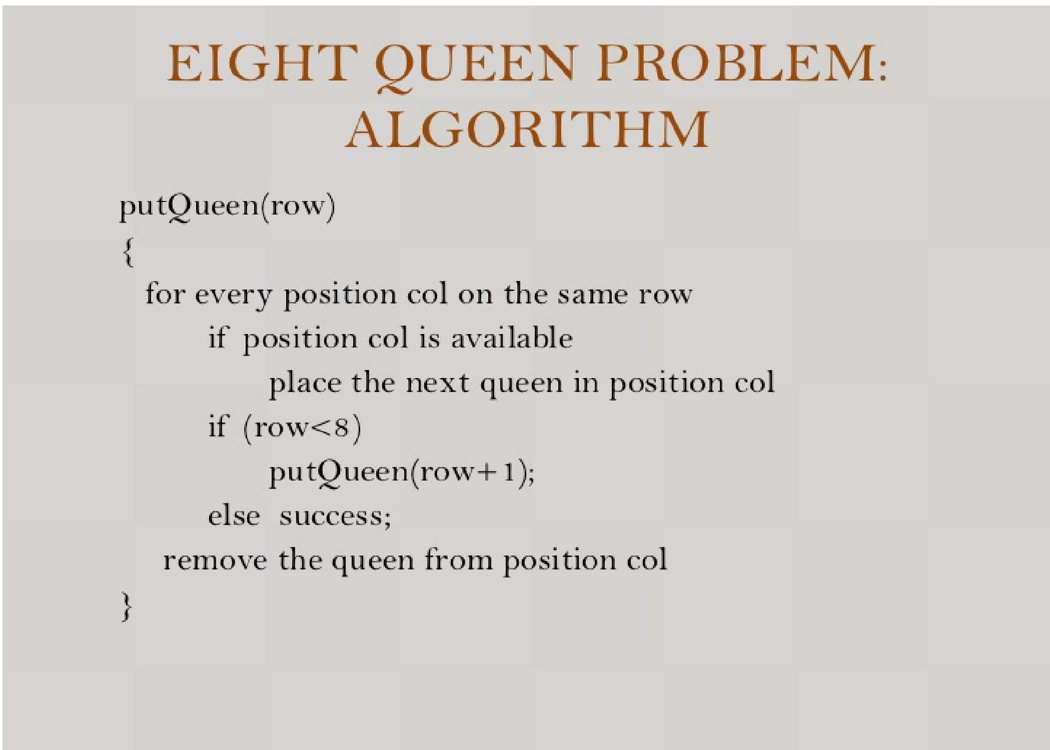

Figure 3: 8-Queens problem using Backtracking

When we carry out backtracking, an easy way to visualize what is going on is a tree that shows all the different possibilities that have been tried. On the board we will show a visual representation of solving the 4 Queens problem (placing 4 queens on a 4x4 board where no two attack one another).

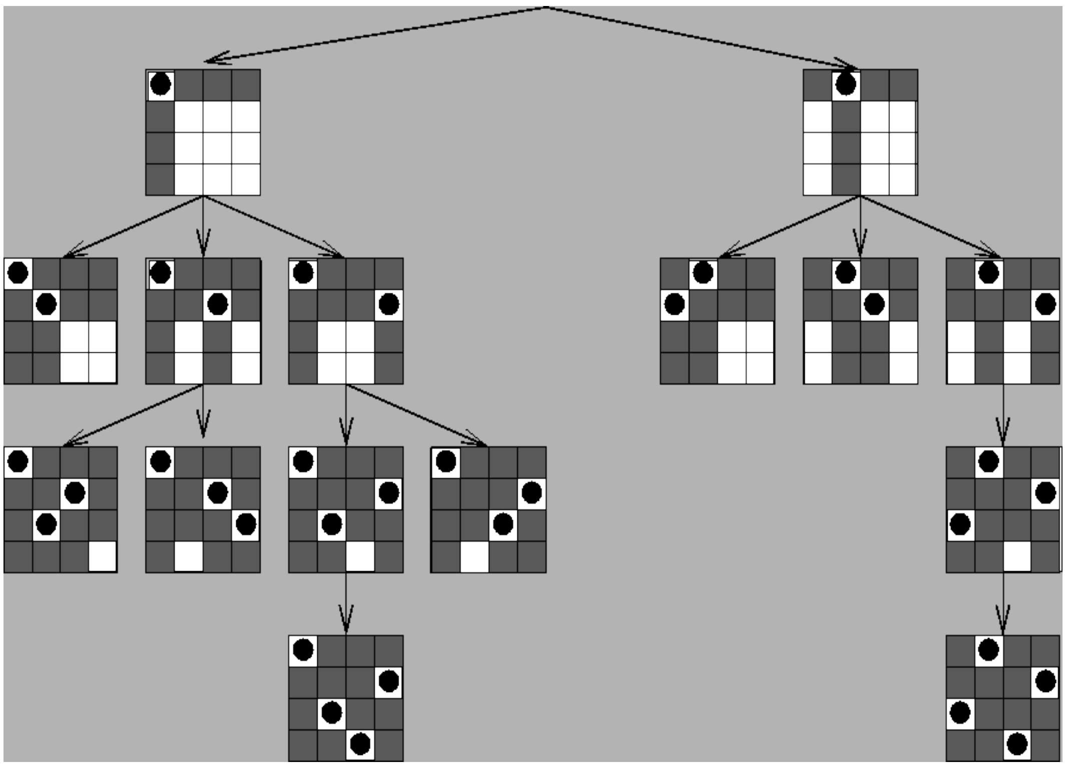

Figure 4: 4-Queens problem Backtracking representation

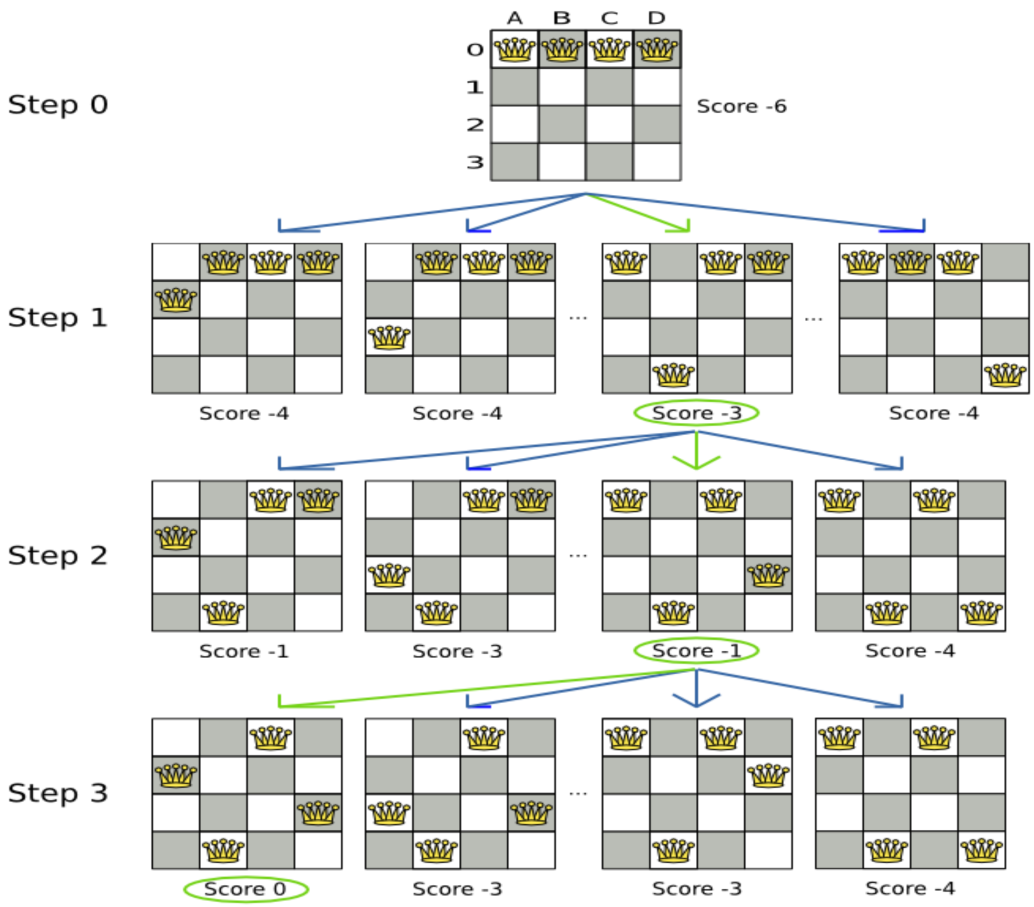

Figure 5: 4-Queens problem Backtracking representation

Mathematical Model

Let S be the system that represents the Eight Queens Algorithm.

Initially,

$$\mathbf{S} = \{\phi\}$$

Let,

$$\mathbf{S} = \{\mathbf{I, O, F}\}$$

Where,
I - Represents Input set
O - Represents Output set
F - Represents Function set

Input set - I:

$$\mathbf{I} = \{\mathbf{X}\}$$

Where,

- X - Represents the input from XML File.

Output set - O:

$$\mathbf{O} = \{\mathbf{C}\}$$

Where,

- L - Represents the configuration of the 8 queens.

Function Set - F:

$$\mathbf{F} = \{F_1, F_2\}$$

Where,

- F_1 - Represents a function that places the queens on a chess board.
  F_1(K, I) -> {T, F}

- F_2 - Represents a function for backtracking in 8 queens.
  F_2(K, N) -> {}

Conclusion

Thus, we have studied and implemented 8 Queens Algorithm using back-tracking in parallel.