Lab-05
To Implement a program to solve the N-Queen problem.

Objectives:
To compute correlation of discrete time signals and study their properties.

Apparatus

- Hardware Requirement
  Personal computer.
- Software Requirement
  Anaconda

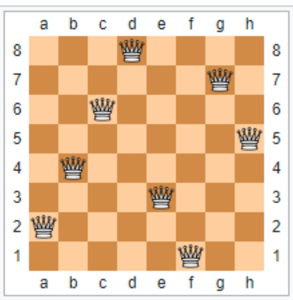
Background:
The **eight queens puzzle** is the problem of placing eight chess queens on an 8×8 chessboard so that no two queens threaten each other; thus, a solution requires that no two queens share the same row, column, or diagonal. The eight queens puzzle is an example of the more general ***n* queens problem** of placing *n* non-attacking queens on an *n*×*n* chessboard, for which solutions exist for all natural numbers *n* with the exception of *n* = 2 and *n* = 3

Constructing and counting solutions:
The problem of finding all solutions to the 8-queens problem can be quite computationally expensive, as there are 4,426,165,368 (i.e., 64C8) possible arrangements of eight queens on an 8×8 board, but only 92 solutions. It is possible to use shortcuts that reduce computational requirements or rules of thumb that avoids brute-force computational techniques. For example, by applying a simple rule that constrains each queen to a single column (or row), though still considered brute force, it is possible to reduce the number of possibilities to 16,777,216 (that is, 88) possible combinations. Generating permutations further reduces the possibilities to just 40,320 (that is, 8!), which are then checked for diagonal attacks.

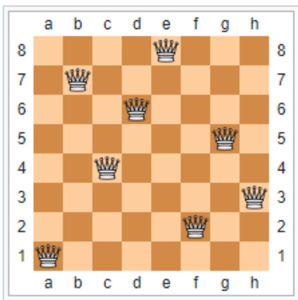
Solutions
The eight queens puzzle has 92 distinct solutions. If solutions that differ only by the symmetry operations of rotation and reflection of the board are counted as one, the puzzle has 12 solutions. These are called fundamental solutions; representatives of each are shown below.

A fundamental solution usually has eight variants (including its original form) obtained by rotating 90°, 180°, or 270° and then reflecting each of the four rotational variants in a mirror in a fixed position. However, should a solution be equivalent to its own 90° rotation (as happens to one solution with five queens on a 5×5 board), that fundamental solution will have only two variants (itself and its reflection). Should a solution be equivalent to its own 180° rotation (but not to its 90° rotation), it will have four variants (itself and its reflection, its 90° rotation and the reflection of that). If n > 1, it is not possible for a solution to be equivalent to its own reflection because that would require two queens to be facing each other. Of the 12 fundamental solutions to the problem with eight queens on an 8×8 board, exactly one (solution 12 below) is equal to its own 180° rotation, and none is equal to its 90° rotation; thus, the number of distinct solutions is 11×8 + 1×4 = 92.

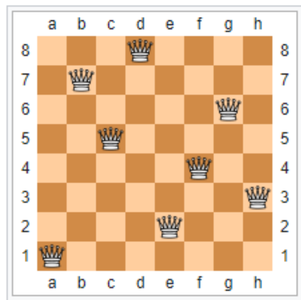
All fundamental solutions are presented below:



Solution 1



Solution 2



Solution 3

Algorithm:

1) Start in the leftmost column

2) If all queens are placed

return true

3) Try all rows in the current column.

Do following for every tried row.

a) If the queen can be placed safely in this row

then mark this [row, column] as part of the solution and recursively check if placing queen here leads to a solution.

b) If placing the queen in [row, column] leads to a solution then return true.

c) If placing queen doesn't lead to a solution then unmark this [row, column] (Backtrack) and go to step (a) to try other rows.

3) If all rows have been tried and nothing worked,

return false to trigger backtracking.

Lab Exercise:

1. Implement the algorithm in python language