Title : Constraint Satisfication Problem

Problem Statement :
Implement crypt-arithmetic problem & n-queens or graph coloring problem.

Objectives :
- To learn and implement n-queens problem.
- To learn the concept of branch and bound & backtracking

Outcomes : We will be able to
- apply branch & bound and backtracking to solve n-queens problem.

S/W & H/W requirements :- OS: Ubuntu/Fedora 20 with Python installed.

Theory :

• Branch and Bound

- Branch and bound is an algorithmic design paradigm for discrete and combinatorical optimization problems.
- It consists of a systematic enumeration of candidate solutions by means of state search space: the set of candidate solutions is thought of as forming a rooted tree. The algorithm explores branches

of this tree, which represent subsets of solution set.

- Before enumerating the candidate solutions of a branch, the branch is checked against upper and lower estimation bounds on the optimal solution and is discarded if it cannot produce a better solution than the best one found so far by the algorithm.

- Backtracking - It can be defined as an algorithmic technique for solving problems recursively by trying to build a solution incrementally, one piece at a time, removing those solutions that fail to satisfy the constraints of the problem at any point of time.

- N-queen problem definition:
The N-Queen is the problem of placing N chess queens on an N×N chessboard so that no two queens attack each other.

P.T.O

Algorithm

1] Start in the leftmost column
2] If all queens are placed
   return true
3] Try all rows in current column
   for every row; do
   (a) If 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 return true.
   (c) If placing queen doesn't lead to a solution then
   unmark this [row, column] (backtrack) and
   go to (a) step to try other rows
4) If all rows have been tried and nothing worked,
   return false to trigger backtracking.

Test cases:

Input board size	Number of solutions
1	1
2	0
8	92
10	724

Conclusion: We have successfully implemented backtracking solution for N-queens problem