N-Queens with DFS

# What is N-queens?

N-queens problem is to place the queen on the board so that no two queens can attack each other,

It’s rule is the queen can move in any straight line , each queen will be located on exactly one row,one column and one diagonal

# DFS:

N-Queens with DFS AI algorithm is to place queens one by one in different columns, starting from the leftmost one , when a queen is placed in a column it checks for clashes with already placed queens .

In the column ,if we find a row for which there is no clashes it mark this row and column as part of the solution , if it doesn’t find such row due to clashes then it backtrack and return false, if it leads to the solution it then returns false,

# 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.

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

return false to trigger backtracking