22/10/24 Week -4

Implement Hill Climbing search algorithm to solve N-Queens problem

Algorithm:

1. Initial state:
   Start with a random configuration of 4 Queens on 4x4 board (each Queen in different column)

2. Heuristic function:
   The Calculate how many pairs of queens are attacking each other.

3. Generate neighbouring boards:
   Move each queen in its column to different rows (one at a time) and calculate new heuristic.

4. Choose the best move:
   If any of the new board has lowest heuristic and choose that new board and move on to that board.

5. Repeat the same steps.

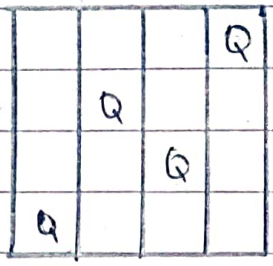
6. Termination:
   If the newly obtaining board has heuristic value of 240. Then return that board as the goal state.
   If no better neighbour, then stop and repeat randomly.

State Space Tree :-

Initial State

fourth position

$x_0 = 3$
$x_1 = 1$
$x_2 = 2$
$x_3 = 0$

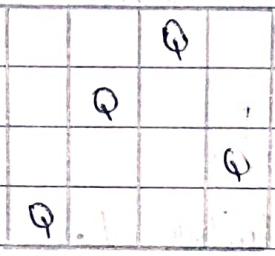


h = 2

swap (x_0, x_2)

Neighbor 1

$x_0 = 2$
$x_1 = 1$
$x_2 = 3$
$x_3 = 0$

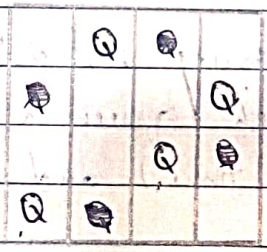


h = 1

swap (x_0, x_1)

neighbour 2

$x_0 = 1$
$x_1 = 3$
$x_2 = 2$
$x_3 = 0$

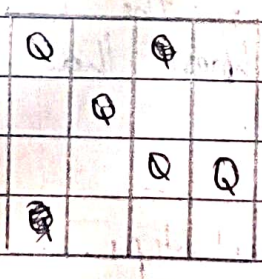


h = 0 1

swap (x_0, x_3)

neighbour 3

$x_0 = 0$
$x_1 = 1$
$x_2 = 2$
$x_3 = 3$

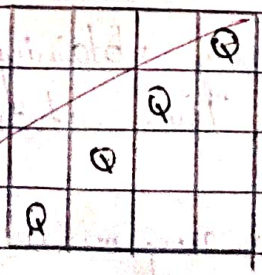


h = 4

swap (x_1, x_2)

neighbour 4

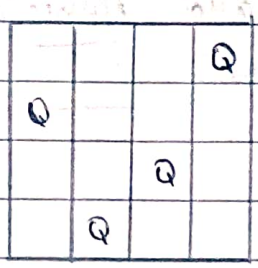
$x_0 = 3$
$x_1 = 2$
$x_2 = 1$
$x_3 = 0$



h = 6

swap (x_1, x_3)

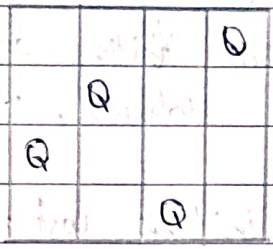
neighbours



$h = 1$

swap (x_2, x_3)

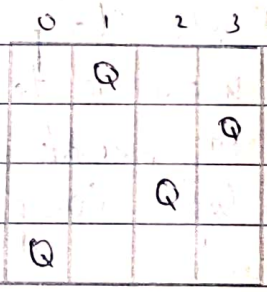
neighbours



$h = 1$

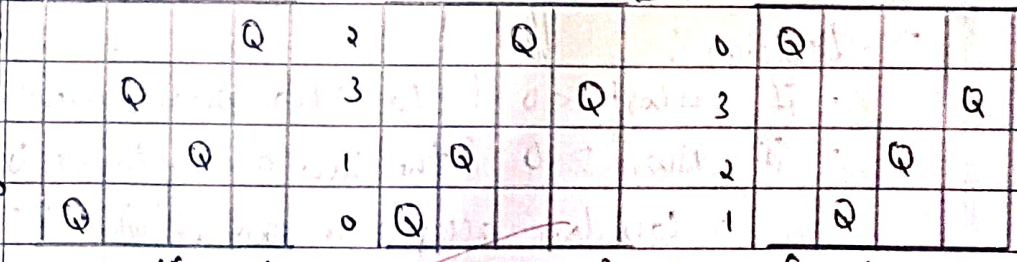
neighbours $(1, 2, 3, 5, 6)$ has $h = 1$

choose neighbour 2.

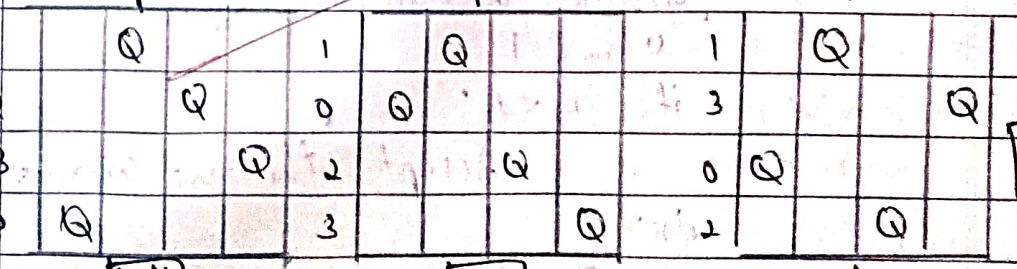


neighbours for neighbour 2

swap (x_0, x_1) $h=2$ swap (x_0, x_2) $h=2$ swap (x_0, x_3) $h=4$



swap (x_1, x_2) swap (x_1, x_3) swap (x_2, x_3)



$h=4$ $h=2$ $h=0$

final (goal state.

22.10