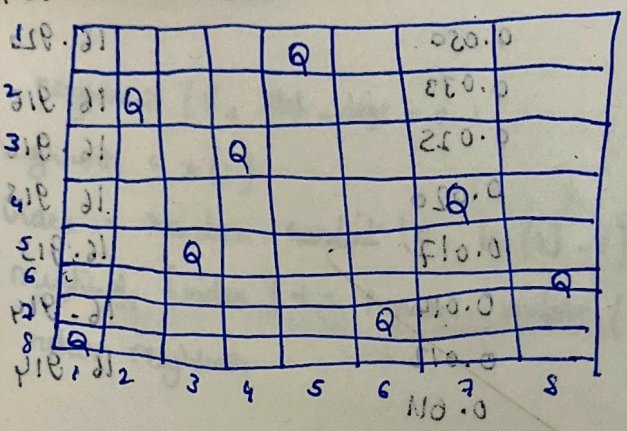
LAB-6 29/10

8 queens -

A* algorithm

Step-1

Construct an 8 × 8 board.



function 8-Queen-Start():

initial = random-positions()

open_list = priority-queue().

open_list.push(initial, heuristics)

while (open_list != 0):

current = open_list.pop-min()

if (heuristic(current) == 0)

return current

// Soln found.

for neighbour in generate(current):

open_list.push(current, heuristic())

return "NO solution".

function heuristic ():

count = 0.

if (same row || same column || same diagonal):

count = 1

return count .

Hill Climbing - 8 Queens

function 8_Queen():

initial = generate_random()

while true :

conflict = heuristic (state)

if (conflict == 0)

return current

// Soln found :

for neighbour in generate (current)

if new_conflict < current_conflict

current = new_state

return "No solution"

function heuristic (State)

count = 0

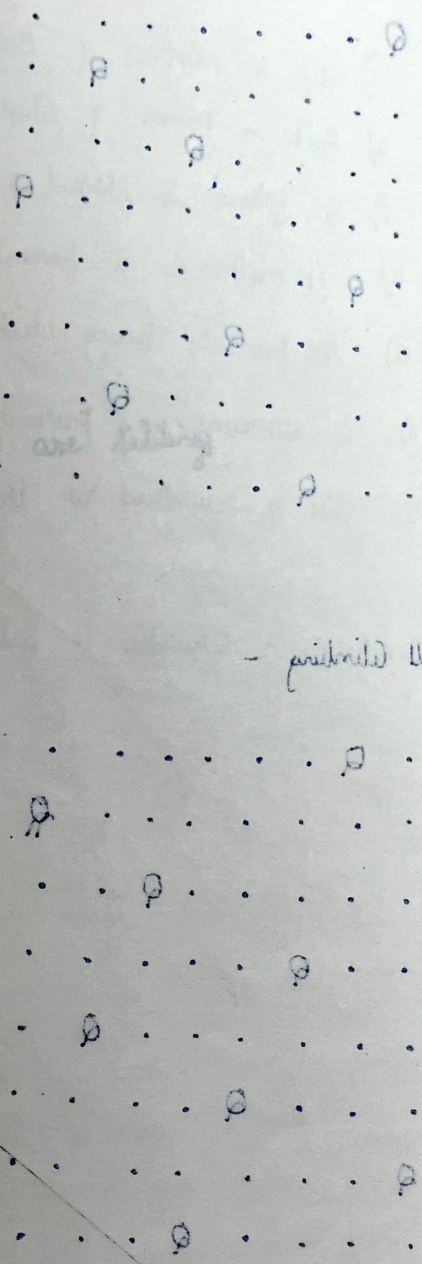
if (same row / same column / same diagonal)

count ++.

return count .

might not find Soln.

Proceed



OUTPUT -

A* - One possible solution

Q . . . . . .
. . . . . Q .
. . . . Q . . .
. . . . . Q
. Q . . . . . .
. . . Q . . . .
. . . . . Q . .
. . Q . . . . .

Hill Climbing -

. Q . . . . . .
. . . . . Q
. . . . . Q . .
. . Q . . . . .
. . . . . Q .
. . . Q . . . .
Q . . . . . .
. . . . Q . . .

No solution found.