algorithm alchalaterat a 8 x8 charsboard to initialize along (O) but up an open del to explore different Configur ations (c) det up a vioited slate its display oil disperent configuration dief & calculate the number of affecting fours that the greens abouted not be in the dame row dame alumn or same diagonal [if state (i) == atate(1) or aus (state(i). column de de lit == y-1 11 diagonal Whan Ottagut: 1 other we increment the variable attacks to determ Tempusiture: 150.0000. Courrender : 100.000 . Compander : 100.000 Evel 3: Orign initial evale to open set for the first a teration. If the node in not visited than forst har it to the green set will push the rook to heaf of Cheronty queues).

neather -heaf puch (open ser Node (now start, g, h) of in the cost to seach the current start in in the not attacks to reach good sign ateps: In the main loop remove the node with the 10 westers + dechs we nave to addressine the next row to place the queen oteps: This halppens up the main book after doing this we openince new state what good calculate h

Hill climbers algorithm Huckin mm. step 1 Place 8 queens nandomly Julieur Transmi soch 2. Find the attacking four such that no gr are peaced in the same row, same weums of same diagonal such 3. Mace & queens randomly on the chentoard state = frandom randinu (0,7) for in sounge (8) current attack : calculate: attack (state) skyy: In the fravious state calculate attacker choose one with few outlacking four archs: uf next-attack> = coverent attack break depolate the current state property state: never state while discent attack = new attack which display board oly a san-3 queens () etale; aufili wir may may cotton published with a state)))) - Most - Dr. the Blow

A star algorithm amport many ay_init_ (aug, atak, 9, h): class Node: dely atare : state oup .9 = 9 dey . 1 = 9+h dep_et_(acel, other): steewen seep. of Lother of de hewershic (date). attacks: 0 for i in range (lun (state)). for i in range (it 1, ten (state)). if state (i) == state (j) or als (state (i) - state == -1-1: altacks +: L orchurn artades oly a-star. 8 queens (); unitial - dtair: tuple ([-1] + 8) Open- Det- [] healing. heapyrush (Ohen - set, Node Cinitial - state, 0, rewritic (initial-state))) united: du-() while often - det! awvent-noxe: heary feapporh (open. set) current - State 2 current_node. Itale if convent-node h == 0 and -1 not in convent-stak; sulva avvers . drap

if avount state in visited: High regulational applications. continue moonus tought. visited add Couverry atak) (hope) consisted molarises pie neset - 8000 = convert - state indesc (-1) of -1 un werent. state else len couvent. state do colin rand(8): new state: list (avoient strate) new-state (next-xow) : co 1 new. Late = tuple (new_state) ug new state not in vuited: 9 = aureunt-node.g+1 h = heuristic (new-atate) heapy heappear (open-out, Node (new state, 9, 12) seturen None (8) Abus 2011 & 10th 102 c out xapq (8): def display (stak): = 1 losox author for row in rangels): A solution for colon sange (8). in state (row) == col: in the and in the contract of deni t = 0 and a consideration of 0 and 0 are a consideration of 0 and 0 are a co State Color attack freent (hine) prent() ouskouds outlist. Coursent solution: a = stor-8, queens() ch sixon exercic if swention. il sense min so fruit ("A auturion:") diplay (dolution) Company District of thousand fruit ("No solution found")

Hill climbing algorithm dy calculate- attacks (steet): for i in range (eun (state)): improbly kaurdom for y in range (it!, cen (atate)): (state [i]. altacks = 0 in range (it!, ean (state). (state (i) - state (i) - state (i) - state (j.) or our (state (i) - state) attach ; 1 xeturn attacks def hill (): state: (sandom. sandint (0,7) for- in sange (8)) ca: calcular-attach(creite) tos-inzande (100): for & in range(8): for e usi xarqu (8): (Ma The Hoad Jule y state (row) ! = w1: : (elugio rini worte nei state (:) (4) = C n. append (n) next-start: muri (n, key; ealculak_altacks) nust-attacks: calculate-attacks (nusor. State) if next-altacks > = current-attacks: ereak State = nest - state . (in the current attacks : new attack vieture state, current-ortade dy diplay (at are): for sin rounge (8): (manipulation) for c in range (8): ig state (8) == c. elvi f: "0" romal hours

lenit: "." print (line) juani () best. Evolution: None bust-attachs -gwar ('ind') attempte 2 100 gor - in rouncy (attempts): 6, a = hell(); if attacks (dust attacks : bust , delution : docution bust-altacks: altacks ig bust - autaous == 0: briak up lest-ableution: fruit ({bust-att-acks 3) diplay (bust-solution) eln: point ("No solution found") output: Bent ablution found Q . · Q ; · · · · · · · Q / · · ...a...

A star algorithm output

Hill climbing algorithm output