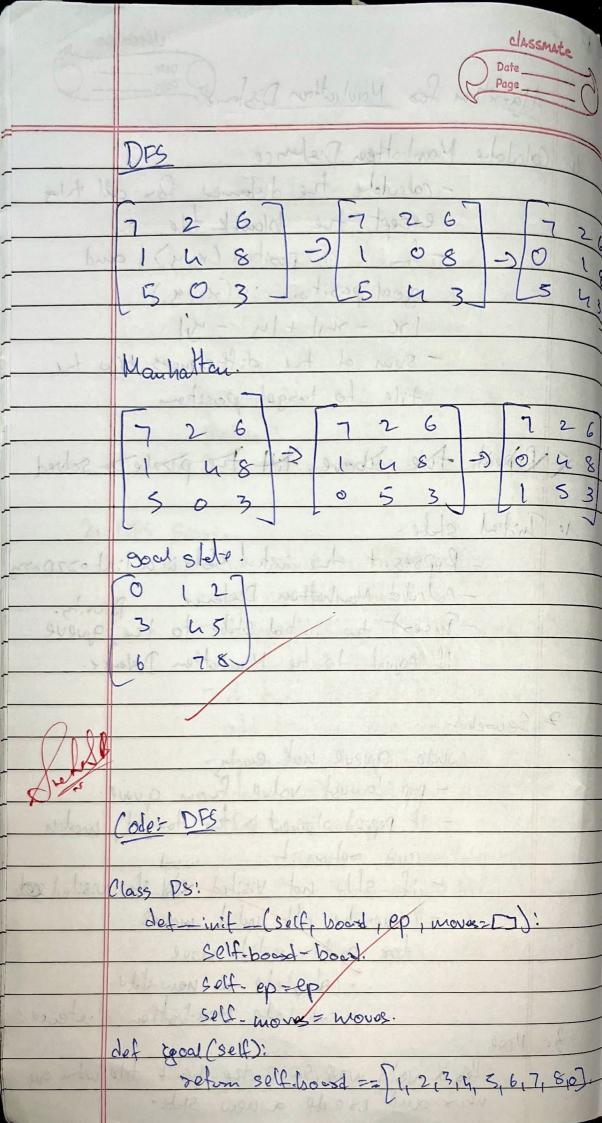
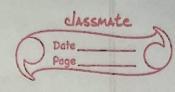


000 classmate Algorithm fox Manhattan Disference Page 4. Calculate Mainhatten Distance - calculate the distance for all they except the blank tile. - for i in position (x,y), and god position is (x, y,) 1x - x1 + 14 - 41 - Sum of the differences blu the tile to target position & Repeat the above, till the poste to solved 10 Initial State. - Represent the inited state in a list o 220 any - Calculat Manhattan Disfered. Porosity. - Jusest the inited offe to the greve if egyed to be Manhotten Distence. 2 Sewoch? While greve not emply. - pop lowest value From grove. - it popped alounds the target, mater - if stole not visited and its visited extended - Creverale all valid moves for each valid more < coloniate a new state. - calculate Mountation distance Move
-fox a valid move, swap tre boant tele widn one
near and create a new state?

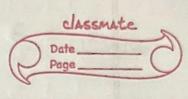




def possmoves (self); X1 q = self-ep movo = D for dt, dy "u[(-1,0),(1,0),(0,-1)(0,1). nx, ny = x + dx, y +dy if 0 1= 4x 63 and 01= 1463: neward = self. board[:] nboar [ no 3 ty], nboard [ nx 3 thy]= 4 board [nx "3 + my], ubrear [n3+4] moves append (in board, (in my)) John move Aet dis(is): Stack, Nissted=[is]: set) while stack: CS = Stack popt) if cs. good(): reform CS-move the Little visited-add (typielcs board) too uboard, nep in (s. possmover): n.slde-puzzle stel ubout nep, cs-moust if type (wonboard) not in visited: Space-appendingtote) Jelin None def pm(board): fos i in souge (0,9,3): pret(boox[i:i+3])

pot()

def main(): ib=[1,2,3,4,0,5,2,8,6] explep = id-indox(o) : 5 = PuzzloState (16, Cep 113, epye3) Broth Duited State: ") pretinder (ib) solution = dfs(is) If Solution: pritti Solution Coud: Los slop in solten:



Code - Man hartfan det md (state, 95) fox i in sange (3): for i in souge (3): if state[i][i] 1 = 0: goodi = [storilli]=) 113 god-j=(state()[j-1)-1.3 List auro f= alos(1-god-j)-1 abol; -90d det getreia (statis 1, j=next((i,j) for in in souge(2) for j h sough) ; f deto(][;]==0) mores [(i-1), (i+1), (i, i-1), (i, j+1) retinifeway stale, i, j, x, y) for x, y in mass if OLEX LB and Olige 3) det sup (state, ii, ji i2, j2)

notale = [sow]: ] for sow in 5 [ Jo] v dado [i][i], nsleteliz[iz] = nsle[iz][iz] usdeCiJCi] refun nelde. dfim (stat, goal, visited = set()): of stale == god: selvan (state) visited, add st& (state) neig = Sosted (getneig (state, Ley=lambde x: m) (v, goal)) too heighbox in noig: if sta (noighbox) not path = of sm (noighbox, god, Visited)
if pathi setum [atale] + path

