10000 10 cy 3) 8 Ruggle Grame good vare : in room A , Room alove: clean { | a 3 4 5 6 2 8 0] Jone Hu Hu Maph 100 8 4 moves = & (1,0) (0,1) (-1,0) (0,-1) } for i in range (3) of (stare(i)(1)/20) for de god of the control range (stare(i)(1)/20) for de god of the according to 3k for j in range(3) goal-i, goal-j = divmod (statelisty) -1,3) mon distance: goal it goal 1 def guncighbours for in sounge(3) Bry in range (3) ", I mosy ni iu may pos_i = it move(o)

fors_j : it move(i) This function gets the neighbouring dements on with freezzic es shuffler this above logic octs the puzzles auranged using markation distance where we calculate the hosizontal and verticle distance of the element in the goal state from unitial star Next in the offs function we take two variables visited and unvisited all the clemens that how been checked and placed in correct order are assigned in visited list and all the clements that are not checked one placed in unwisited his we do their as that the same element is not checked in the new state again and again fire interior interior

funda collections GLOAL SIATES ,12,5,1) [4,8,6] 10351 curr- atak: { } goal-state (2) stack funn (own olar) MOVES: 1 moves: 0. (-1,0). ((0,1) left = (0,1) column (1-,0) (r;0) Sight (0,-1) old mornination dust down: (1,0 distance = 0 for i in range (3): fount (mover) You in somaria): of state [1] [7] = 0: gan in goal y = samod ataraillist -1,3) distance + I air (i-grain) + als (y-grain) def is good description): o this - 1901 == survey houses · (3=- (7/10/2-b) ; ex more in movies is in the mercial of the rest. i

```
from collections import deque
  GOAL STATES
      (1,2,3),
      [4,5,6],
      13,8,07
 MOVES:
    (-1,0).
    (1,0),
    (0,-1),
     (0,1)
dy manhatan-distance (state):
 distance = 0
  for i in range (3):
    for y'in rounale(2):
      y statefilij][=0:
         goal-i, goal-j'= airmod ataresissis-1,3)
         duitance + = als (i-goal-i) + als (y'-goal-j')
    oceturn distanc
def - is - goal. State (state):
    sectiven atout == GOAL-STATE
des get-neighbours (state).
    ne ighbors = []
    for i in range (3):
       for in range (3);
        ij (statelil(j.)==0).
           for move in moves:
             ni, nj: it move(o), y't move(1)
          uj 0 (: ni 23 and 0 <= nj 23:
new state: [row]: ] por row in state)
new state(i)(i), new state (ni)(nj); new state(ni)((nj), new state
neighbors. appliend (new stare)
surious relighbors
```

def also (state): tu fluit queue : olcque ([(state, [state])]) morphical (611,01 viaited z set () 18,1,21 while queue: 13,5,41 cs, fr = queue. prophet() [6,8,2] ef eis-goal state (cs): (1,0,1) (+,1,H) outwork ej duple (man(duple, cs)) in visited: (0,5,5) [8,0,2] continue visited add (type (map (tupe, cs))) (6,1,3) queue append (meigneux, pods fit (neigneux)) for neighbor in gct. neighbors (es) outurn Non (1,2,3) [8 11 12] initial state - P [2,2,6] 14,1,37, (3,0,5) (8,2,4) (7,2,6), [6,3,1] (5,8,0) 12,0,27 freath = of (initial state) (4,8,4) fruit ("dolution found") for state in h. for row in state: frant (xow) four () frant ("No debution found")

Bulfart (liste : still) docution found: (0,1,3) [4,1,3] [4,2,6] (3,2,6) (3,8,4) (5,8,0) (1,0,3) (4,1,3) [4,2,6] (3,2,6) a Chedle super (7,5,8) (2,0,8) (1,2,3) and from super- Little (4,1,3) (3,7,6) (0,5,8) (2,5,8) [4113] (1,2,3) 90,2,67 14,5,67 (3,5,8) (31018) [1,2,3) P415,67 (0,8,E)

1+1+0+++3+0+6+2=8
total moves 8