

8-puzzle problem using DF3 and MD Lab-3 > 1. Using DEs Initial Setup We assume initial array and final array (both 3×3) are defined with random g integer from 0-8 placed (0 = empty space) > Algorithmse using DFs: at dep ofs (boostart, end): book Stack ? [] Stack append (hash (start)) while stackful! = end; valid-mover = getMover (stack[-1]) Stack of] for move in valid - moves: det des (start, end start) Stack = 1]; Visited = 1]

Stack append (start) while stack (-1] != end: valid-more, get Movy (Stack (-1)) for more in valid - mover !

elu:

& apply Move nen-board = board. copy () HA por (new board [new ps), new board, [zero Pos + move]

if new-board not in visited: Stack append (new board) visited append (new-board)

while Stack-popl) != new-board: Confinsa

Continue ruturn stackfitj

	Date
	Page
	Utility functions:
	- det get Mover (zoop board, zur Pas):
7	valid-mover 7
-	
	for more in valid tractimore) (0)
	1 0 5 2000 Por (0) + move (1) C= 3.
	and 0 <= zero Por [1] + move [1] Caz.
	and 0 2 games
•	Continue
<u> </u>	elx:
	valid-movel. nemore (move)
	AL TOTAL
\rightarrow	Using Manhettan Distance:
	del mod Start en 42, words
	ONLY = 80 17; VISITED = 13
	quevo. append (start)
	visited append (stary)
	Valid-mover = get Mover (2000 books) Pucks
while que	vilo) for move in valid-mover!
] =	end new board = board. copy ()
	new board of Swap (new-board groß)
	new-board (zeroloi)
	if new-board not in visited:
	que append (new board)
	golde applied (tras toute)
	visited append (new-board)
	greve sont (key = get MD ()
	electric de la constant de la consta
	queve: 70p(0)
	La Carlo Car
-	
-	

		Date		
	> Utility function:			
	- def get MD (board):	Comment of the second section of the	The state of the s	
	for i in narge (3)	and the second s		٠.
	for in range	3):		
	C += (i -)	poord (i)	(j) 113)
	return C	board	11(3) 7.	3)
	THEOLOGIC			
	Eg:	2		
	Start = 2 3 4 2.			
	7 7 1 MD -	19	2+1+	ų.
,	8 96 0			
	Possible Mover 2 2 3 4 8		3 (
	⊕ 5 = 1 = 1 = 1 = 1 = 1 = 1 = 1 = 1 = 1 =		8 6	
	MD = 19		MD =	
6	Gro with O	•		
,	Possible Mover?			Χ
	2 3 4 2 5	<u>ų </u>	2 3	,
	0 8 6 8 7			6 0
	MD 7 17 MD = 1			
	Gro with Da			
	DSB MARKET MARKE	D/ 1 2	4.84	1.
Or.		V		111
,	The state of the s			

	Page
	Program:
,	import heapq
	class Pozzle:
	detinit- (self):
	Self board - L
	(1, 2, 3)
	(4,0,5)
	[7,8,6]]
	Self. ende [
	(0, 4, 8)
	[-4, M, 6] (4, M, 6)
	(6,7,8) [7,8,0])
-	
	det get Mover (self, board):
	gero-pos - self zero-index (board)
	moves = ((0,1), (1,0), (0,1), (1,0))
1.	valid-moves = []
	for move in moves
	if OC= zero-poslo) + move(o) (? av
	OC: 200-pos (1) + muve (1) <3!
	valid-mover apperd (move)
	netvan valid-movers
	det zero-index (self, board):
	for i in range (2)
	for j in range (3):
	if board Ci) (j) = · 0:
	noturn (1, j)
	TO T
	det bhash (self, board).
	of neturn -tuple (map (tuple, board))
	The Company Boards

	def display: (self, board):
	for ls in board:
	print (+ Us)
	det des (seef):
	Stack - (7)
	visited = ()
	Stack. append (Self. Loard)
	visited. append (self. bhash (self. board))
,	while Stack!
	top + Stack (-1)
	if Self. bhash (SHF. board Sey end):
-	Valid-mover, get Mover (top)
	added - False
	for move in valid-move:
	new-board (Prop (:) for now in top)
	zeroPo) = Self zero-index (new-board)
	newPos = [graPos (o) + move(o),
	ZeroPos [1] + move [1]
	new-board [new Por (0)) (New Por (1)),
	new board [Pero Pa (07) [zero Pos (17), =
	new-board (zeo Por 10) (zero Por 10),
	new-board [new Pos(o)] (new Pos (1))
	if Seef bhash (new-board) not in visited
	Stack append (new-board)
	visited. append (seef. bhank (new-bond)
	added - True
	breck
	if not added:
	Stack.pop()
	ashila stable:
	self. display (stack. pop(0)); print ("").

	The state of the s	Date	
and the state of t	•	W .	
	0 . 10		
	C= Poze(1)		,
	@ point ("DFS: ")		
	c.dfs()		
	Output:		
	DFS:		
	1 2 3		
	U O Talleton and the second		
	7 8 6		
	1 2 3		
	4 10		
	1-8-6-1-2	,	
1	1 2 3		
. ,	Li Tob		
	7 8 0	•	
		-	
the state of the			-
		<u> </u>	
	a large of the late of the lat	*	
	transfer to the second		
19.10	A Property of the state of the		
`	ų.		. /
	The state of the s		