IBM18CS057

Naman Agarwal.

Global Variables

Removed
At code
stage

- puzzle = [[1,2,3],[-1,4,6],[7,5,8]]
- final = [[1,2,3],[4,5,6],[7,8,-1]]

Structure of Variables

- Arr [] → [puzzle, int: depth, int: hvalue]
- duplicateUp, duplicate down, duplicateLeft, duplicateRight.
  → [puzzle, int: hvalue]
- Assend[] → [duplicate up]

def depth():   → Gives hValue

    Arr = [[puzzle, 0, compare()]]

    if depthNonEfficient (Arr)

        return

    else

        #Not found.

def depth NonEfficient (Arr)

    temp = depthEfficient (Arr)   → True if found.

    if temp

        print (Arr)

    else

        return false.

def depthEfficient (Arr)

    current = Arr[len(Arr) -1]

    if depth conn[1] >= 15.

        return false.

    if conn [2] == 0

        return true.

A. Agarwal

(Ann, ~~functions~~) = check all (Ann)

puzzle = Ann [len(Ann)-1][0]

depthEfficient (Ann) ⟶ Recursive

def checkAll():

   i; j → find blank

   if i != 0
       # move up and save in duplicate Up &
   if j != 0   left
       # move down and save in duplicate ~~Down~~ left.
   if i != 2
       # move Down and save in duplicate Down.
   if j != 2
       # move Right and save it duplicate up Right.

   Ascend ⟵ Store all duplicates in ascending
                   order of hvalue.

   Ann. append (Ascend [0])

   return (Ann)

A.Agarwal.