Sudoku Solver

Def function to solve puzzle(takes x,y and count value):

Define hold value to track number of times count value passes valid number check

Increase count value by 1

Define variables for 3X3 box check (x1,x2,y1,y2)

If count value = 10 set it to 0 and reset last changed box in puzzles array to 0

For loop from 0-8

If puzzleArray[x][y]==0:

If puzzleArray[x][i] and puzzles Array[i][y] does not equal count

Increase hold value by 1

#End if puzzle array

#end if puzzle array

#end for loop

If x>3:

Set x1 to 3

Set x2 to 0

Elif x>6 and 3>x:

Set x1 to 6

Set x2 to 3

Else

Set x1 to 9

Set x2 to 6

#end if x

If y>3:

Set y1 to 3

Set y2 to 0

Elif y>6 and 3>y:

Set y1 to 6

Set y2 to 3

Else

Set y1 to 9

Set y2 to 6

#end if y

For loop from x2 to x1

For loop from x2 to x1

If puzzles array[i][j]<>count

Hold increases by 1

#end if puzzle

#end for loop

#end for loop

If hold==18:

puzzles array[x][y]=count

Increase y by 1

If y==9:

Increase x by 1

If x<> 9:

Set y to 0

Else

Return to end function

#end if 3

#end if y

#end if hold

If x<>9:

Recursive call to function (gives x,y,and count value)

#end if x

#end def solve

Define count variable, x and y and z variables all set to 0

Define x and y values for 3x3 box checking

Define variavle to hold number of check passes

Define variable to hold a stop value

Define arrays to hold puzzle file, 2D puzzle file, and puzzle w locks

Open sudoku file read to puzzle array

Convert into a 2D array w loops

(Checking if puzzle can be solved)

(Row and column check)

For loop form 0-8 (x)

For loop 0-8 (y)

For loop 0-8 (k)

if puzzle[x][y]==puzzle[x][k] and puzzle[x][y]<>0 and k<>y:

Set stopper to 1

elif puzzle[x][y]==puzzle[k][y] and puzzle[x][y]<>0 and k<>x:

Set stopper to 1

#end if puzzle

#end for loop

#end for loop

#end for loop

(3x3 box check)

For loop 0-8 (z)

If z module 3 is 0 and z is not 0

Set x2 to -3

Set x1 to 0

Increase y1 by 3

increase y2 by 3

#end if x

For loop x2 - x1 (x)

For loop y2 - y1 (y)

For loop x2 to x1 (i)

For loop y2 to y1 (j)

If puzzle[x][y] is not 0 not puzzle[x][y] doesn’t equal puzzle[i][j] and x is not i and y is not j

Set stopper to 1

#end if puzzle

#end for loop

#end for loop

#end for loop

#end for loop

#end for loop

If stopper equals 1 tell user that puzzle can not be solved

Else tell user puzzle is solvable

(Locking filled boxes)

For loop 0-8

For loop 0-8

If puzzle[x][y] equals 0

puzzleLock[x][y] is set to “O”

Else

puzzleLock[x][y] is set to “L”

#End if puzzle

#end for loop

#end for loop

(calling function)

If stop value is 0

Call solve function

#end if stop

(Printing solved sudoku board)

For loop 0-8 (x)

If x%3 is 0

Print horizontal separating lines

#end if x

For loop 0-8 (y)

If y%3==0

Print vertical separating line

Print puzzle[x][y]

#end for loop

#end for loop