# The code is written and run for swaps(i,j) for their in-order occurrence in the set Pd (**case-1**)

#programming Language - python

import math

def generateSwapsList(n, d) :

Pd = []

for x in range(1,n+1):

m = n-d+1

Pd.append((x,x))

for i in range(d, m):

if (x+i) <= n:

Pd.append((x,x+i))

return Pd

def generatePermutationsSet(base, swaps, l) :

swapsCopy = []

permutationsSet = []

if l <= 0:

return [tuple(base)]

swapsCopy = swaps.copy()

for swap in swaps:

p = swapPositions(base[:], swap)

swapsCopy.remove(swap)

permutationsSet.extend(generatePermutationsSet(p, swapsCopy, l-1 ))

return permutationsSet

def swapPositions(base, swap):

if swap[0] == swap[1]:

return base

temp = base[swap[0]-1]

base[swap[0]-1] = base[swap[1]-1]

base[swap[1]-1] = temp

return base

d\_max = 4 #d=1,2,3,4

l\_max = 9 #l=1,2,3,4,5,6,7,8,9

sigmaPermutation = [1,2,3,4,5,6,7,8,9] #[1,2,3,4,.......,n]

towPermutation = [9,8,7,6,5,4,3,2,1]

n = len(sigmaPermutation)

# we have assumed that d <= n/2 and l <= number of element in swaps (Pd set)

for d in range(1,d\_max+1):

swaps = generateSwapsList(n, d)

#takes n and d as input parameters and return swaps set Pd

for l in range(1,l\_max+1):

sigmaSet = set(generatePermutationsSet(sigmaPermutation, swaps, math.floor(l/2)))

towSet = set(generatePermutationsSet(towPermutation, swaps, math.ceil(l/2)))

setIntersection = sigmaSet & towSet

if not setIntersection:

print("NO", end =" ")

else:

print("YES", end =" ")

print('\n')

#end of code

The results obtained on running the above program for d = 1,2,3,4 and l = 1,2,3,4,5,6,7,8,9 over **σ** ( sigma-permutation) = 1,2,3,4,5,6,7,8 and ***τ* (**tow=permutation) = 9,8,7,6,5,4,3,2,1 are as shown below in the form of 4x9 matrix

NO NO NO YES YES YES YES YES YES   
  
NO NO NO NO NO YES YES YES YES   
  
NO NO NO NO NO YES YES YES YES   
  
NO NO NO NO NO NO NO NO NO

Transforming results into a tabular format, rows are for D, columns are for L

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| D\L | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 1 | NO | NO | NO | YES | YES | YES | YES | YES | YES |
| 2 | NO | NO | NO | NO | NO | YES | YES | YES | YES |
| 3 | NO | NO | NO | NO | NO | YESY | YESY | YES | YES |
| 4 | NO | NO | NO | NO | NO | NO | NO | NO | NO |