Find the String

Given two integers N and K, the task is to find the string S of minimum length such that it contains all possible strings of size N as a substring. The characters of the string should be from integers ranging from 0 to K-1.

Example 1:

Input:

N = 2, K = 2

Output:

00110

Explanation:

Allowed characters are from 0 to k-1 (i.e., 0 and 1). There are 4 string possible of size N=2 (i.e "00", "01","10","11") "00110" contains all possible string as a substring. It also has the minimum length.

Example 2:

Input:

N = 2, K = 3

Output:

0010211220

Explanation:

Allowed characters are from 0 to k-1 (i.e., 0, 1 and 2). There are total 9 strings possible of size N, given output string has the minimum length that contains all those strings as substring.

Your Task:

You don't need to read input or print anything. Complete the function findString( ) which takes the integer N and the integer K as input parameters and returns the string.

Note: In case of multiple answers, return any string of minimum length which satisfies above condition. The driver will print the length of the string. In case of wrong answer it will print -1.

Expected Time Complexity: O(KNlogK)

Expected Space Complexity: O(KNN)

Constraints:

1 ≤ N ≤ 4

1 < K < 10

1 < KNN < 106

Solution:

#User function Template for python3

class Solution:

def findString(self, N, K):

def \_\_init\_\_(self):

self.a=None

def dfs(n,k,string,ans):

if n==0:

ans.add(string)

return ans

for i in range(k):

ans=dfs(n-1,k,string+str(i),ans)

return ans

def find(ele,string):

vis.add(ele)

if len(vis)==(K\*\*N):

self.a=string

return True

for y in graph[ele]:

if y not in vis and find(y,string+y[-1]):

return True

vis.remove(ele)

return False

ans=set()

ans=dfs(N,K,'',ans)

graph={}

for i in ans:

for z in range(K):

ele=i[1:]+str(z)

if ele in ans and ele!=i:

if i in graph:

graph[i].append(ele)

else:

graph[i]=[ele]

vis=set()

for i in ans:

if find(i,i):

return self.a

return self.a

