Coin Change - Permutations - 2

1. You are given a number n, representing the count of coins.
2. You are given n numbers, representing the denominations of n coins.
3. You are given a number "amt".
4. You are required to calculate and print the permutations of the n coins (same coin can be used again any number of times) using which the amount "amt" can be paid.

Note -> Use the code snippet and follow the algorithm discussed in question video. The judge can't force you but the intention is to teach a concept. Play in spirit of the question.

A number n
n1
n2
.. n number of elements
A number amt

1 <= n <= 30
0 <= n1, n2, .. n elements <= 20
0 <= amt <= 50

3
2
3
5
7

2-2-3-.
2-3-2-.
2-5-.
3-2-2-.
5-2-.

```
def coinsChangePermutationII(coins, amount):
    idx = 0
    ans = []
    ssf = ''
    visited = [False] * len(coins)
    helper(coins, amount, ans, ssf, visited)
    return ans



def helper(coins, amount, ans, ssf, visited):
    if amount == 0:
        ans.append(ssf[:-1])
        return
    if amount < 0:
        return
    for i in range(len(coins)):
        helper(coins, amount - coins[i], ans, ssf + str(coins[i]) + '-', visited)


print(coinsChangePermutationII([2, 3, 5], 7))
```