Combinations Sum

Question: Given a set of candidate numbers (C) and a target number (T), find all unique combinations in C where the candidate numbers sums to T.

The same repeated number may be chosen from C unlimited number of times.

Note:

All numbers (including target) will be positive integers.

Elements in a combination (a1, a2, … , ak) must be in non-descending order. (ie, a1 ≤ a2 ≤ … ≤ ak).

The solution set must not contain duplicate combinations.

For example, given candidate set 2,3,6,7 and target 7,

A solution set is:

[7] ; [2, 2, 3]

Solutions:

```
class Solution:
    def combinationSum(self, candidates, target):
        candidates.sort()
        res=[]
        self.DFS(candidates,target,0,res,[])
        return res

    def DFS(self,candidates,target,start,res,intermedia):
        if target==0:
            res.append(intermedia)
            return
```

```
        for i in range(start,len(candidates)):
            if target<candidates[i]:
                return
            self.DFS(candidates,target-
candidates[i],i,res,intermedia+[candidates[i]])


print( Solution().combinationSum([2,3,6,7],7) )
```