

找出所有相加之和为  $n$  的  $k$  个数的组合。组合中只允许含有 1 - 9 的正整数，并且每种组合中不存在重复的数字。

说明：

- 所有数字都是正整数。
- 解集不能包含重复的组合。

示例 1:

输入:  $k = 3, n = 7$

输出: `[[1,2,4]]`

示例 2:

输入:  $k = 3, n = 9$

输出: `[[1,2,6], [1,3,5], [2,3,4]]`

分析，该题为回溯算法的题目，回溯给人想起了递归的深度搜索，可能符合条件的，进行搜索，不符合的，则回到上一个路径，进行查找，最终得到可能的值。代码如下。

```
class Solution {
public:
    void dfs(vector<vector<int>> &result, vector<int> nums, map<int, int>
    &bj, int sum, int k, int n, int j) {
        if (k == 0) {
            if (sum == n) {
                result.push_back(nums);
            } else {
                return;
            }
        }
        for (int i = j; i < 10; i++) {
            if (bj[i] == 0 && sum + i <= n) {
                bj[i] = 1;
                nums.push_back(i);
                dfs(result, nums, bj, sum + i, k - 1, n, i + 1);
                nums.pop_back();
                bj[i] = 0;
            }
        }
    }

    vector<vector<int>> combinationSum3(int k, int n) {
        vector<int> nums;
        vector<vector<int>> result;
        map<int, int> bj;
        dfs(result, nums, bj, 0, k, n, 1);
        return result;
    }
};
```

```
    }  
};
```