C++, Clean Backtracking, 0 ms, beats 100%, Easy-to-understand

[lllllll2](https://leetcode.com/lllllll2/)

870

4450

Sep 12, 2020

C++

C

Backtracking

**Explanation**  
The key to this question is backtracking, that is, generate possible combinations and see which one fulfills the conditions.  
We choose a number for 1,2..kth position in ascending order, and for each position, we only iterate through [current\_number, 9].

**Code**

class Solution {

public:

vector<vector<int>> ans;

void f(vector<int>& cur, int cnum, int k, int n) {

if(n < 0 || cur.size() > k) return;

if(n == 0 && cur.size() == k) {

ans.push\_back(cur);

return;

}

for(int i=cnum; i<=9; ++i) {

cur.push\_back(i);

f(cur, i+1, k, n-i);

cur.pop\_back();

}

}

vector<vector<int>> combinationSum3(int k, int n) {

vector<int> cur;

f(cur, 1, k, n);

return ans;

}

};