/\*\*

\* @param {number[]} candidates

\* @param {number} target

\* @return {number[][]}

\*/

//该题如何防止出现重复值是个难点，想了很久才想出来。需要好好琢磨一下

let Ans = [];

var maketree = function(i,cur,ans,candidates,target)

{

console.log(ans);

if(cur == target)

{

Ans.push(ans);

return ;

}

while(cur+candidates[i]<=target && i<candidates.length)

{

let temp = ans.slice(0,ans.length);

temp.push(candidates[i]);

maketree(i+1,cur+candidates[i],temp,candidates,target);

i++;

while(candidates[i]==candidates[i-1]) //这一块卡了很久没想出来该这样去重

{

i++;

}

}

}

var combinationSum2 = function(candidates, target) {

candidates = candidates.sort((a,b)=>a-b);

let ans = [];

Ans = [];

let set = new Set();

let order = 0;

for(let i = 0;i < candidates.length;i++)

{

//if(i==0 || candidates[i]!==candidates[i-1])

if(!set.has(candidates[i]))

{

set.add(candidates[i]);

ans = [candidates[i]];

maketree(i+1,candidates[i],ans,candidates,target);

}

}

return Ans;

};