1. Subsets II

Medium

Given a collection of integers that might contain duplicates, **nums**, return all possible subsets (the power set).

**Note:** The solution set must not contain duplicate subsets.

**Example:**

Input: [1,2,2]  
Output:  
[  
 [2],  
 [1],  
 [1,2,2],  
 [2,2],  
 [1,2],  
 []  
]

**Solution**

没有想明白去重

class Solution {  
public:  
 vector<vector<int>> subsetsWithDup(vector<int>& nums) {  
 vector<int>tmp;  
 set<vector<int> >ans;  
 dfs(nums, tmp, ans, 0);  
 vector<vector<int> >res;  
 for(auto it = ans.begin(); it != ans.end(); ++it)res.push\_back(\*it);  
 return res;  
 }  
 void dfs(vector<int>& nums, vector<int>&tmp, set<vector<int> >&ans, int index){  
 if(index == nums.size()){  
 vector<int>tmpp = tmp;  
 sort(tmpp.begin(), tmpp.end());  
 ans.insert(tmpp);  
 return;  
 }  
 dfs(nums, tmp, ans, index + 1);  
 tmp.push\_back(nums[index]);  
 dfs(nums, tmp, ans, index + 1);  
 tmp.pop\_back();  
 }  
};