

78. Subsets

Description

Hints

Submissions

Discuss

Solution

Pick One

Given a set of **distinct** integers, *nums*, return all possible subsets (the power set).

Note: The solution set must not contain duplicate subsets.

Example:

```
Input: nums = [1,2,3]
Output:
[
  [3],
  [1],
  [2],
  [1,2,3],
  [1,3],
  [2,3],
  [1,2],
  []
]
```

```
/*
 * 这道题目用到了树的回溯法
 * 要求给出所有可能的子集，直接用动态规划的方法求解，因为每一次backtrack的时候都有解，所以在每次
 * backtrack的时候将结果写入即可
 */
public class L78 {
    public List<List<Integer>> subsets(int[] nums) {
        List<List<Integer>> resList = new ArrayList<>();
        Arrays.sort(nums);
        List<Integer> list=new ArrayList<>();
        backtracking(resList, list, nums, 0);
        return resList;
    }

    public void backtracking(List<List<Integer>> resList,List<Integer> list,int[] nums,int start) {
        //这儿的list是一个，指针
        resList.add(new ArrayList<>(list));
        for(int i=start;i<nums.length;i++){
            list.add(nums[i]);
            backtracking(resList, list, nums, i+1);
            list.remove(list.size()-1);
        }
    }
}
```

例 [1, 2, 3]

list =

list = [1]

pop, list = [1]

pop 3 list = [1]

list = [1, 2]

pop 2 list = [1]

list = [1, 3]

list = [1, 2, 3]

pop 3 list = [1, 2]

list = [2]

list = [2]

list = [2, 3]

list = [3]