



[LeetCode] 78. Subsets 子集合

Given a set of distinct integers, S , return all possible subsets.

Note:

- Elements in a subset must be in non-descending order.
- The solution set must not contain duplicate subsets.

For example,

If $S = [1, 2, 3]$, a solution is:

```
[
  [3],
  [1],
  [2],
  [1, 2, 3],
  [1, 3],
  [2, 3],
  [1, 2],
  []
]
```

这道求子集合的问题，由于其要列出所有结果，按照以往的经验，肯定是要用递归来做。这道题其实它的非递归解法相对来说更简单一点，下面我们先来看非递归的解法，由于题目要求子集合中数字的顺序是非降序排列的，所有我们需要预处理，先给输入数组排序，然后再进一步处理，最开始我在想的时候，是想按照子集的长度由少到多全部写出来，比如子集长度为0的就是空集，空集是任何集合的子集，满足条件，直接加入。下面长度为1的子集，直接一个循环加入所有数字，子集长度为2的话可以用两个循环，但是这种想法到后面就行不通了，因为循环的个数不能无限的增长，所以我们必须换一种思路。我们可以一位一位的网上叠加，比如对于题目中给的例子 $[1, 2, 3]$ 来说，最开始是空集，那么我们现在要处理1，就在空集上加1，为 $[1]$ ，现在我们有自己 $[]$ 和 $[1]$ ，下面我们来处理2，我们在之前的子集基础上，每个都加个2，可以分别得到 $[2]$, $[1, 2]$ ，那么现在所有的子集合为 $[]$, $[1]$, $[2]$, $[1, 2]$ ，同理处理3的情况可得 $[3]$, $[1, 3]$, $[2, 3]$, $[1, 2, 3]$ ，再加上之前的子集就是所有的子集合了，代码如下：

解法一：



公告



(请关注下方微信公众号，并留言跟博主联系)

Github同步地址，欢迎star♡

github.com/grandyang/leetcode

搜索【shua2sum】或扫描二维码
关注微信公众号【刷尽天下】



使用方法：

- 回复数字【0】随机推送一道题。
- 回复区间【1 - 919】内任意数字推送对应的题目。
- 回复关键字 例如【Two Sum】推送对应的题目。
- 回复【all】推送题目汇总列表。
- 回复【other】推送相关总结帖。
- 回复任意文字跟博主留言交流^ . ^

昵称： Grandyang

园龄： 7年11个月

粉丝： 1124

关注： 36

+加关注

2020年3月						
日	一	二	三	四	五	六
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31	1	2	3	4
5	6	7	8	9	10	11

搜索

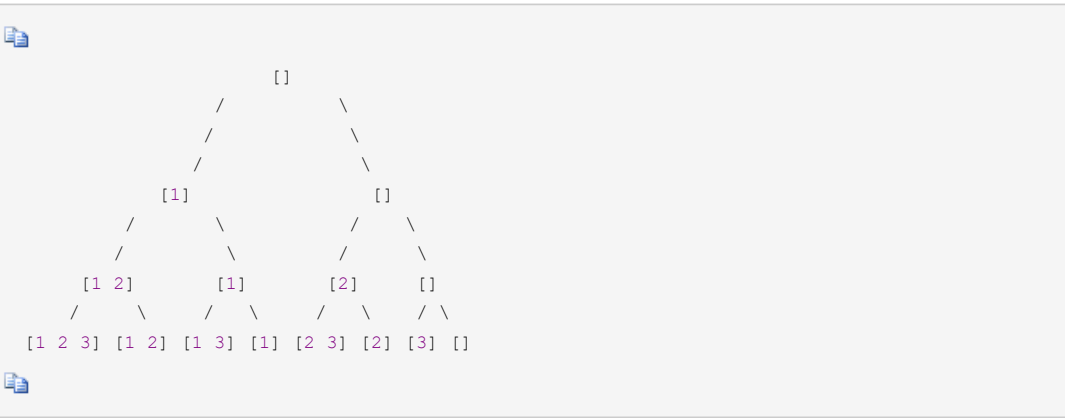
最新随笔

```
class Solution {
public:
    vector<vector<int>> > subsets(vector<int> &S) {
        vector<vector<int>> > res(1);
        sort(S.begin(), S.end());
        for (int i = 0; i < S.size(); ++i) {
            int size = res.size();
            for (int j = 0; j < size; ++j) {
                res.push_back(res[j]);
                res.back().push_back(S[i]);
            }
        }
        return res;
    }
};
```

整个添加的顺序为：

- []
- [1]
- [2]
- [1 2]
- [3]
- [1 3]
- [2 3]
- [1 2 3]

下面来看递归的解法，相当于一种深度优先搜索，参见网友 [JustDolt](#) 的博客，由于原集合每一个数字只有两种状态，要么存在，要么不存在，那么在构造子集时就有选择和不选择两种情况，所以可以构造一棵二叉树，左子树表示选择该层处理的节点，右子树表示不选择，最终的叶节点就是所有子集合，树的结构如下：



解法二：

```
class Solution {
public:
    vector<vector<int>> > subsets(vector<int> &S) {
        vector<vector<int>> > res;
        vector<int> out;
        sort(S.begin(), S.end());
        getSubsets(S, 0, out, res);
        return res;
    }
};
```

- 1.[LeetCode] 935. Knight Dialer 骑士拨号器
- 2.[LeetCode] 933. Number of Recent Calls 最近的调用次数
- 3.Solve Error: Could not find the certificate xxxx.com. at ServerlessCustomDomain.<anonymous>
- 4.[LeetCode] 934. Shortest Bridge 最短的桥梁
- 5.[LeetCode] 932. Beautiful Array 漂亮数组
- 6.[LeetCode] 931. Minimum Falling Path Sum 下降路径最小和
- 7.[LeetCode] 930. Binary Subarrays With Sum 二元子数组之和
- 8.[LeetCode] 929. Unique Email Addresses 独特的邮件地址
- 9.[LeetCode] 928. Minimize Malware Spread II 最大程度上减少恶软件的传播之二
- 10.[LeetCode] 927. Three Equal Parts 三个相等的部分

赞
助

积分与排名

积分 - 2972207
排名 - 16

随笔分类

- 3D Visualization(12)
- Algorithms(8)
- Amazon Web Service(3)
- C/C++, Java, Python(34)
- CareerCup(150)
- CUDA/OpenCL(1)
- Digital Image Processing(3)
- Entertainment(6)
- GTK+/VTK/ITK/FLTK(20)
- IOS(7)
- LaTex(3)
- LeetCode(906)
- LintCode(101)
- MatLab(10)
- Maya / 3ds Max(10)
- MySQL(2)
- Node.js / JavaScript(8)
- OpenCV(37)
- Point Grey Research(11)
- Qt(49)
- Software/Tools(3)
- Useful Links(34)

随笔档案

- 2020年3月(1)
- 2020年2月(5)
- 2020年1月(3)
- 2019年12月(2)
- 2019年11月(4)
- 2019年10月(9)
- 2019年9月(8)
- 2019年8月(10)
- 2019年7月(8)

```
    }
    void getSubsets(vector<int> &S, int pos, vector<int> &out, vector<vector<int> > &res) {
        res.push_back(out);
        for (int i = pos; i < S.size(); ++i) {
            out.push_back(S[i]);
            getSubsets(S, i + 1, out, res);
            out.pop_back();
        }
    }
};
```



整个添加的顺序为：

- []
- [1]
- [1 2]
- [1 2 3]
- [1 3]
- [2]
- [2 3]
- [3]

最后我们再来看一种解法，这种解法是 CareerCup 书上给的一种解法，想法也比较巧妙，把数组中所有的数分配一个状态，true 表示这个数在子集中出现，false 表示在子集中不出现，那么对于一个长度为n的数组，每个数字都有出现与不出现两种情况，所以共有 2^n 中情况，那么我们把每种情况都转换出来就是子集了，我们还是用题目中的例子, [1 2 3] 这个数组共有8个子集，每个子集的序号的二进制表示，把是1的位对应原数组中的数字取出来就是一个子集，八种情况都取出来就是所有的子集了，参见代码如下：

	1	2	3	Subset
0	F	F	F	[]
1	F	F	T	3
2	F	T	F	2
3	F	T	T	23
4	T	F	F	1
5	T	F	T	13
6	T	T	F	12
7	T	T	T	123

解法三：



```
class Solution {
public:
    vector<vector<int> > subsets(vector<int> &S) {
        vector<vector<int> > res;
        sort(S.begin(), S.end());
        int max = 1 << S.size();
        for (int k = 0; k < max; ++k) {
            vector<int> out = convertIntToSet(S, k);
```

- 2019年6月(13)
- 2019年5月(16)
- 2019年4月(14)
- 2019年3月(10)
- 2019年2月(12)
- 2019年1月(10)
- 2018年12月(8)
- 2018年11月(19)
- 2018年10月(9)
- 2018年9月(6)
- 2018年8月(8)
- 2018年7月(11)
- 2018年6月(10)
- 2018年5月(11)
- 2018年4月(13)
- 2018年3月(15)
- 2018年2月(14)
- 2018年1月(17)
- 2017年12月(12)
- 2017年11月(16)
- 2017年10月(29)
- 2017年9月(21)
- 2017年8月(10)
- 2017年7月(12)
- 2017年6月(21)
- 2017年5月(26)
- 2017年4月(18)
- 2017年3月(22)
- 2017年2月(23)
- 2017年1月(13)
- 2016年12月(26)
- 2016年11月(30)
- 2016年10月(30)
- 2016年9月(24)
- 2016年8月(40)
- 2016年7月(31)
- 2016年6月(33)
- 2016年5月(30)
- 2016年4月(70)
- 2016年3月(32)
- 2016年2月(32)
- 2016年1月(25)
- 2015年12月(3)
- 2015年11月(36)
- 2015年10月(43)
- 2015年9月(51)
- 2015年8月(46)
- 2015年7月(45)
- 2015年6月(29)
- 2015年5月(28)
- 2015年4月(42)
- 2015年3月(55)
- 2015年2月(61)
- 2015年1月(27)
- 2014年12月(8)
- 2014年11月(27)
- 2014年10月(35)
- 2014年9月(5)

最新评论

- 1. Re:LeetCode All in One 题目讲解汇总(持续更新中...)

赞助

```
        res.push_back(out);
    }
    return res;
}
vector<int> convertIntToSet(vector<int> &S, int k) {
    vector<int> sub;
    int idx = 0;
    for (int i = k; i > 0; i >= 1) {
        if ((i & 1) == 1) {
            sub.push_back(S[idx]);
        }
        ++idx;
    }
    return sub;
}
};
```



Github 同步地址:

<https://github.com/grandyang/leetcode/issues/78>

类似题目:

- [Subsets II](#)
- [Generalized Abbreviation](#)
- [Letter Case Permutation](#)

参考资料:

- <https://leetcode.com/problems/subsets/>
- <https://leetcode.com/problems/subsets/discuss/27288/My-solution-using-bit-manipulation>
- <https://leetcode.com/problems/subsets/discuss/27278/C%2B%2B-RecursiveterativeBit-Manipulation>
- [https://leetcode.com/problems/subsets/discuss/27281/A-general-approach-to-backtracking-questions-in-Java-\(Subsets-Permutations-Combination-Sum-Palindrome-Partitioning\)](https://leetcode.com/problems/subsets/discuss/27281/A-general-approach-to-backtracking-questions-in-Java-(Subsets-Permutations-Combination-Sum-Palindrome-Partitioning))

[LeetCode All in One 题目讲解汇总\(持续更新中...\)](#)

分类: [LeetCode](#)

好文要顶

关注我

收藏该文

Grandyang
关注 - 36
粉丝 - 1124
[+加关注](#)

« 上一篇: [\[LeetCode\] 92. Reverse Linked List II 倒置链表之二](#)
» 下一篇: [QComboBox 和 QSpinBox 使用方法](#)
posted @ 2015-03-02 17:29 Grandyang 阅读(20940) 评论(6) 编辑 收藏

评论列表

#1楼 2016-02-14 01:57 hupeikang1

我有个土方法:
可以把树构建成

3

推荐

0

反对

ios商店怎么搜不到这款应用了?

--我呆不了一个月了

2. Re:[LeetCode] 23. Merge k Sorted Lists 合并k个有序链表
@Grandyang 我用python写了按顺序合并, 可以通过, 但耗时很长, 5936ms~...

--lypbendif

3. Re:[LeetCode] 351. Android Unlock Patterns 安卓解锁模式
@CV学习者 把m=2, n=2带入, 得到的结果是 56。1可以到达五个数字, 分别是 2, 4, 5, 6, 8。2可以到达七个数字, 分别是 1, 3, 4, 5, 6, 7, 9。5可以到达八个数字, 分别是 1, 2...

--Grandyang

4. Re:[LeetCode] 351. Android Unlock Patterns 安卓解锁模式
@CV学习者 是有这样的解锁方式啊, 没错啊, 1可以直接到6或8。...

--Grandyang

5. Re:[LeetCode] 351. Android Unlock Patterns 安卓解锁模式
@Grandyang 不会出现1 6 9这样的解锁方式吗? jumps[1][6]==0, 不就进递归里面了? ...

--CV学习者

赞助

阅读排行榜

1. LeetCode All in One 题目讲解汇总(持续更新中...)(788181)
2. [LeetCode] 1. Two Sum 两数之和(116966)
3. [LeetCode] 15. 3Sum 三数之和(68819)
4. Manacher's Algorithm 马拉车算法(63131)
5. [LeetCode] 4. Median of Two Sorted Arrays 两个有序数组的中位数(54641)
6. [LeetCode] 5. Longest Palindromic Substring 最长回文子串(54423)
7. [LeetCode] 3. Longest Substring Without Repeating Characters 最长无重复字符的子串(48294)
8. [LeetCode] 10. Regular Expression Matching 正则表达式匹配(44020)
9. Qt qDebug() 的使用方法(39520)
10. [LeetCode] 2. Add Two Numbers 两个数字相加(37545)

评论排行榜

1. LeetCode All in One 题目讲解汇总(持续更新中...)(154)
2. [FlyCapture2] Bumblebee XB3 Save Images to Three AVI Files (L

""
1 2 3
12 13 22 23 32 33
123 133 223

然后剪枝:
if (oneResult.Count == 0 || oneResult[oneResult.Count - 1] < nums[i])

```
1 public class Solution
2 {
3     public List<List<int>> Subsets(int[] nums)
4     {
5         var result = new List<List<int>>();
6         Array.Sort(nums);
7         int depth = 0;
8         SubsetsDFS(nums, depth, new List<int>(), result);
9
10        return result;
11    }
12
13    private void SubsetsDFS(int[] nums, int startIndex, List<int> oneResult, List<List<int>> result)
14    {
15        result.Add(new List<int>(oneResult.ToArray()));
16
17        for (int i = startIndex; i < nums.Length; ++i)
18        {
19            if (oneResult.Count == 0 || oneResult[oneResult.Count - 1] < nums[i])
20            {
21                oneResult.Add(nums[i]);
22                SubsetsDFS(nums, startIndex + 1, oneResult, result);
23                oneResult.RemoveAt(oneResult.Count - 1);
24            }
25        }
26    }
27 }
```

支持(0) 反对(0)

#2楼 2016-02-14 02:37 hupeikang1

或者:
SubsetsDFS(nums, Math.Max(startIndex + 1, i + 1), oneResult, result);

StartIndex 和 i 取最大的那个。

也可以剪掉

支持(0) 反对(0)

#3楼 2017-01-08 18:33 leostc

博主你好，请问各个方法的时间和空间复杂度是多少。
我感觉解法1和2时间复杂度应该是一样的，为啥leetcode上算法2结果明显比1快呢

支持(0) 反对(0)

#4楼 [楼主] 2017-01-14 06:41 Grandyang

@ leostc
我感觉都是O(2^n)吧
解法一和解法二在我机子跑的时间基本一样啊

支持(0) 反对(0)

#5楼 2017-11-19 13:41 hiimdaosui

这题为什么要sort?

支持(2) 反对(0)

#6楼 2018-03-29 22:50 普兒

eft, Center and Right) 大黄蜂立体
相机保存捕获的视频到左中右三个
不同的文件(35)
3. [LeetCode] 4. Median of Two So
rted Arrays 两个有序数组的中位数
(35)
4. [LeetCode] 1. Two Sum 两数之
和(32)
5. LeetCode Binary Search Summ
ary 二分搜索法小结(32)

推荐排行榜

- 1. LeetCode All in One 题目讲解汇
总(持续更新中...)(90)
- 2. Manacher's Algorithm 马拉车算
法(17)
- 3. Reward List 赏金列表(8)
- 4. [LeetCode] 1. Two Sum 两数之
和(7)
- 5. [LeetCode] 94. Binary Tree
der Traversal 二叉树的中序遍历
- 6. [LeetCode] 15. 3Sum 三数之
(6)
- 7. [LeetCode] 289. Game of Life
生命游戏(6)
- 8. [LeetCode] 407. Trapping Rain
Water II 收集雨水之二(5)
- 9. LeetCode Binary Search Summ
ary 二分搜索法小结(4)
- 10. [LeetCode] 287. Find the Dupli
cate Number 寻找重复数(4)

赞助

@ hiimdaosui
因为题目要求非降序排列吧?

支持(0) 反对(0)

刷新评论 刷新页面 返回顶部

注册用户登录后才能发表评论，请 [登录](#) 或 [注册](#)，访问 [网站首页](#)。

【推荐】超50万行VC++源码: 大型组态工控、电力仿真CAD与GIS源码库
【活动】腾讯云服务器推出云产品采购季 1核2G首年仅需99元
【推荐】独家下载 | 《大数据工程师必读手册》揭秘阿里如何玩转大数据
【推荐】独家下载电子书 | 前端必看! 阿里这样实现前端代码智能生成

亿速云

香港云服务器

免备案·SSD存储·新一代CPU

29元起

12元红包

相关博文:

- [LeetCode \(78\) : 子集](#)
- [LeetCode Subsets I& II——递归](#)
- [算法 - 子集划分](#)
- [代码题 \(18\) 子集](#)
- [\[LeetCode\] Subsets II 子集合之二](#)

» [更多推荐...](#)

阿里专家五年方法论总结! 技术人如何实现职业突破?

最新 IT 新闻:

- [宜家开通天猫旗舰店 初期只向上海、江苏等地提供服务](#)
- [广告分一成, 内购只抽15%, 华为向Google Play发起总攻](#)
- [2020, 旅游行业的翻牌与重启](#)
- [三星显示器与越南政府沟通 要求免除700名韩国工程师的强制隔离](#)
- [17年历史, 这个RCE漏洞已影响数个Linux系统](#)

» [更多新闻...](#)

赞助