

Pinterest

题目都是地里见过的，详情可以参考这个帖子，有题目的详细汇总

<https://www.1point3acres.com/bbs ... read&tid=448141>

- 60min面试，前15min聊一聊做过的project，后45min Coding
- 面试时间24/7随便你选，还有真人和你聊，而且CodePad的界面和平时用的 scripting app（比如我用的sublime text）还是很接近的，会高亮关键词，甚至可以自动填充，还允许你网上去查API（查之前要和面试官说清楚）
- 唯一需要注意的就是，coding的每一道题是需要你过test case的，就是要能实际跑出来的，有bug要现场debug，main函数也要自己写，这和狗家的意识流还是有不一样的地方的
- 面试为1小时，前15分钟盘问简历，45分钟code。需要execute code。
- domain计数subdomain
domain, subdomain click counts (like[sports.yahoo.com, 100, ...])
返回domain 和subdomain 上点击的次数。输入是一个array, 每个element 是 domain address 和点击的次数。
求longest continuous visit records, given two users browser history list.
- longest common substring
- 1. dp[longest common suffix] — $O(m*n)$, s: $O(m*n)$ <https://www.geeksforgeeks.org/longest-common-substring-dp-29/>
<https://www.geeksforgeeks.org/longest-common-substring-space-optimized-dp-solution/>
 $O(m*n)$, s: $O(2n)$
- 2. Suffix tree <https://www.geeksforgeeks.org/suffix-tree-application-5-longest-common-substring-2/>
 $O(m+n)$
- // Suppose we have some input data describing a graph of relationships between parents and children over multiple generations. The data is formatted as a list of (parent, child) pairs, where each individual is assigned a unique integer identifier.

// For example, in this diagram, 3 is a child of 1 and 2, and 5 is a child of 4:

```
// 1  2  4
// \ /  /\
//  3 5  8
//  \/\  \
//   6 7  9
```

// Write a function that takes this data as input and returns two collections: one containing all individuals with zero known parents, and one containing all individuals with exactly one known parent.

. 一亩-三分-地, 独家发布

// Sample output (pseudocode):

// [

// [1, 2, 4], // Individuals with zero parents

// [5, 7, 8, 9] // Individuals with exactly one parent

//]

// Write a function that, for two given individuals in our dataset, returns true if and only if they share at least one ancestor.

// Sample input and output:

// parentChildPairs, 3, 8 => false

// parentChildPairs, 5, 8 => true

// parentChildPairs, 6, 8 => true

经典题：找零度&一度node, 公共祖先, 最远祖先, 三小问.

1 2 3

/ \ / \

4 5 6

 \

 7

输入是int[][] input, input[0]是input[1]的parent, 比如 {{1,4}, {1,5}, {2,5}, {3,6}, {6,7}}会形成上面的图, done

第一问是只有0个parents和只有1个parent的节点

第二问是 两个指定的点有没有公共祖先. From 1point 3acres bbs

第三问是就一个点的最远祖先, 如果有好几个就只需要输出一个就好, 举个栗子, 这里5的最远祖先可以是1或者2, 输出任意一个就可以

○ 给了三个list, 第一个是purchase的userid list, 第二个是广告click记录, 每一条是

[IP address, AD], 第三个是userid, IP对照表, 然后问对每条ad来说, # of purchase/ # of click。还有些别的假设: 每个用户只有一次purchase, 每个用户只有一个ip-go

- We have a library interfacing with S3 keys, providing an S3Key.next() function which returns the next 8 characters in the file. Empty string is returned when it reaches the end.

For example, with below contents

'This is the first line.\nThis is the second line.'

```
S3Key.next() -> 'This is '  
S3Key.next() -> 'the firs'  
S3Key.next() -> 't line.\n'  
S3Key.next() -> 'This is '  
S3Key.next() -> 'the seco'  
S3Key.next() -> 'nd line.'  
S3Key.next() -> ''
```

The files can be very large and therefore impractical to fully download before working with them.

Write a wrapper around S3Key with function next_line() which produces the next line in a file.

```
sln.next_line() => 'This is the first line.\n'  
sln.next_line() => 'This is the second line'
```

—> 用string存一下buffer (search \n, \n后的东西存成buffer), 每次先看看buffer里面有没有\n

- Given 2 input arrays of integers, unsorted. 从两个array各拿出一个数相加, 成为一个pair, 找出largest k pairs

2, 3, -1, 0, 5

1, 4, 2

k = 5

--> (5, 4), (5, 2), (3, 4), (5, 1), (2, 4)

<https://www.geeksforgeeks.org/k-maximum-sum-combinations-two-arrays/>

—> 自定义Pair<Sum, index i, index j>, 用max heap

装，每次pop一个，把 $\langle \text{new sum}, l + 1, j \rangle$ and $\langle \text{new sum}, l, j + 1 \rangle$ 扔进去，复杂度 $O(N \log N) + O(K \log K)$

- LC 34

- LC 354 没时间写，只说了 n^2 dp解法

- 133 clone graph

- 一共三问

1. 有一个由 $(-1, 0)$ 组成的board: -1 是wall, 0 是可以走的。给你一个cell = 0 的position, 返回它上下左右四个方向的是 0 的neighbors。就是很直接，看它四个方向的neighbor是 0 就返回就行了。。

[[0, -1, 0, 0],

[0, 0, -1, 0],

[0, 0, 0, -1]]

2. 还是这个board，给你一个cell = 0 的end position，判断这个board的其他的所有的 0 是否能reach到它。比如上面的这个，假如end position是左上角那个 0 . 就要返回false，因为右上角那3个 0 reach不到它。

3. [[0, -1, 0, 0],

[0, 1, -1, 0],. From 1point 3acres bbs

[0, 1, 0, 0]] board里面有1，为treasures，给一个start position和end position。

要找到一条shorted path, from start position to end position. 并且经过所有的1.不能走重复的格子

第二题是用BFS + visited做的，第三题当时只有10分钟了，我的想法就是直接DFS找所有start 到end的路径，然后找出过了所有1的，然后再比较最短的。就说了说思路，我后来还挺担心第三轮没写完代码会有什么问题。。不过面试官说you will get all credits of what you have done.所以我觉得这家只要做出来前两道题就差怒多了，还要他家问简历里的项目问的还挺细的，我觉得需要好好准备下

- 找长方体和shape

找长方体和shape的那三题，分析时空复杂度。

第二题：找int[][] matrix里的用0表示的box的范围。范围可以用左上，右下或者类似的表示。

第一小问：只有一个box

第二小问：有多个box。这一问应该和面试官讨论有两个box重合的情况。

第三小问：box是不规则的。每个box的定义变成这个box的所有的点的集合

第一小问只有一个矩形box；第二问有多个，互相不连着；第三问形状不规则。

- meeting room 2 变种，每个interval有weight，overlap的部分，weight相加
- 给字典和一个字符串，输出最短的substring使其包括所有字典里的char（李特口原题变种，原题字典里的char会重复，面试的时候不会重复）

- topological sort['bca','be','ec'] 每个string里面前面的字母比后面的priority高，然后把所有的排序，比如这个就输出'beca'

follow up: 什么Input情况下程序会有问题。回答 比如有circle之类的

LZ这个string数组是 拓扑排序后的数组还是说是用拓扑排序去做这道题

输出错了 c应该在e前面

不用看两个string之间的顺序，只看string 里面每个字母的顺序，比如ec, e在c前面。be, b在e前面

- 给一些<studentID, course name>，返回每两个学生之间上的一样的课。

eg. ("1", "A"), ("2", "B"), ("5", "E"), ("1", "B"), ("2", "A"), ("2", "E")

返回 ("1", "2") -> ["A","B"]

("1", "5") -> []

("2", "5") -> ["E"]

2. 给一些<course, course>代表prerequisite，只有一条可行的路径，返回中间的那节课。

eg. ("A", "C"), ("B", "D"), ("D", "A"), ("G", "E"), ("C", "F"), ("E", "B")

唯一路径为 G->E->B->D->A->C->F

返回D

3. 给一些<course, course>代表prerequisite，返回每条路径中间的那节课。

eg. ("A", "B"), ("B", "D"), ("E", "B"), ("E", "C"), ("C", "F"), ("E", "F")

路径为 ABD, EBD, ECF, EF

返回B,C,E

- 第一题：计算机

第一小问：只有加，减，数字

第二小问：在之前的基础上有了括号

第三小问：在此基础上有variable。有的可以map上数字，有的不行。return一个最简化后的string

calculator那道= = 带variable的那个我还没搞明白= =