BST 的 add, find 和 delete 函数
定义一个 BST，然后写 insert 或者 add 方法.就着上面的结构，怎样确定一棵树是不是 BST. 然后他打了两个例子，让说这个算法是怎么跑的
第五轮是给一个 BST，可以有 duplicate values，找出出现次数最多的 value，还问了一个把 aabbbcc 压缩成 a*2b*3c*2，再解压缩，原来的字符串里可能也有数字和*
determine if a node is in a BST
BTS, 先写了一个 recursive，5 分钟搞定，然后让我写一个不是 recursive 的
build a BST from an array
大数据的查找问题；BST 和 Hash Table 的适用场合以及性能比较；实现任意一个字符串向 float 数的转变。
given one BST, find the Kth minimum value
构建特殊堆 A rooted binary tree with keys in its nodes has the binary search tree property (BST property) if, for every node, the keys in its left subtree are smaller than its own key, and the keys in its right subtree are larger than its own key. It has the heap property if, for every node, the keys of its children are all smaller than its own key. You are given a set of n binary tree nodes that each contain an integer i and an integer j. No two i values are equal and no two j values are equal. We must assemble the nodes into a single binary tree where the i values obey the BST property and the j values obey the heap property. If you pay attention only to the second key in each node, the tree looks like a heap, and if you pay attention only to the first key in each node, it looks like a binary search tree.Describe a recursive algorithm for assembling such a tree

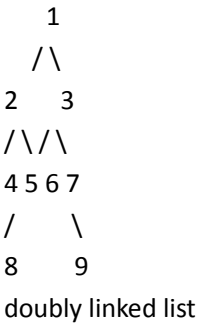
in-order 遍历；实现一个成员函数，用来返回下一个遍历的节点。
clone graph，用了一个很实际的问题去描述，说得很复杂，不过最后就是实现 clone graph 的变种。

给定一个 binary search tree， 知道到第 k 大的数。
实现一个 familiy tree,每个 node 有多个 parent 多个 children， 这个 parent tree 里面可能有环。除了 constructor destructor，要实现一个 CommonAncestor(node1, node2)函数，判断两个 node 是否是亲戚
1.给两个四分树，求两个图重叠的 1 的个数
2.怎么 continuous deploy
3.run length representation 的合并
find k-th (start from 0) node in binary search tree。 先说了一下 inorder 遍历的方法，然后让 improve 写了个 binary search 的版本。
给了一个基本的 binary search tree 写 2 个 function： first，return 最小的 node; next，return 下一个大的 Node（不是最大，是 next big node）最后，用这两个 function，print the whole tree out and give out runtime
现在有 父母，夫妻，孩子，父母与孩子，兄弟姐妹这些关系。给 2 个人，怎么查出有没有血缘关系。

写一个图的数据结构（图是一个 DAG）
根据你的数据结构，写一个递归遍历图算法，和非递归遍历算法。
如何改变你的数据结构使得： 当多个线程调用你的遍历算法时，每个线程都可以 visit 所有节点，且每个线程都只访问图中每个节点一次。

family tree（check whether two people have some ancestor）

merge two quadtree
Google : Convert a binary tree into a double linked spiral list
Write a program to convert binary tree in to doubly linked list such that doubly linked list represents the spiral order of given tree using recursion, Inplace(no Extra Space except recursion memory stack)
given pointer to head of binary tree



doubly linked list represent
head-> 1 2 3 7 6 5 4

binary tree serialization/deserialization 每个 node 存放一个 string. serialization 之后返回一个 string, deserialization 这个 string 必须返回和原始 tree 一样结构的 tree .后来在网上看，得知 null node 也存储的话, preorder 或者 postorder 就可以唯一确定一个 tree 了

给一个 set，里面是一堆 pair，每个 pair 里是两个 string，一个 first，一个 second，假设这堆 pair 能够构成一个树状结构，按照一定的格式打印这棵树
first-second 关系类似 paretnt-child 关系
eg
set: (a, b) (b, c) (a, d) (d, e) (d, f) (d, g)
树状结构是 root = a, root.left = b, root.right = d blah blah
打印结果：[space] 就是一个空格
a
[space]b
[space][space]c
[space]d
[space][space]e
[space][space]f
[space][space]g

给一个整形数组，找离数组的平均值最近的数。写完后问如果该成一个可能随时加数进去的 list，怎么找最近的数。分别说说怎么实现 add(int)和 findNearestAvg()。我想了想说大概用 list 或者用 tree 维持一个 sorted list 然后再二分查找，但是感觉不能同时保证 add 和 find 都是 logN 的。。然后他觉得是对的就下一题了。。

怎么找二叉树的最深节点。说 dfs 就能弄，记录深度值，每次比较存最大的那个。

线索化二叉树
给定一个树内的节点的指针，在不能返回整个遍历结果的情况下输出中序遍历的下一个节点，顺利的写出了程序但是也犯了些不必要的小错误。
红黑树
Of the pairs of words in the dictionary that have no letters in common, find one that maximizes the product of the words' lengths.
cat
dog
feed
pull
space
pair = word1, word2
length = word1.size() * word2.size()
given a text file and a list of string, find the max length of strings that combined from the given list of string in the text file. Each character in the given list of string can be used only once.
给你一个很大的字典。一对词如果不 share 任何字母，比如 dog, cat 不 share 字母，而 dog, boy 就 share 一个 o，则是 interesting pair.找出所以 interesting pairs 中长度乘积最大的 pair.输出这个乘积。

超级 prime 定义为所有前缀是 prime 的数， 比如 239 (2 , 23, 239 are all prime numbers) ,given N 打印所有长度为 N 的超级 prime
decide if a number is prime

print all the primes below a certain number
求一个数 n 的所有 prime factor。比如输入 96 输出 2 2 2 2 2 3。
问题让我打印出质数。
我问打印多少或者到哪个数为止，他说不停打印就是了，感觉就是一股老子就是要让 cpu 飞转的气势。。。
这种数学类问题我完全没有准备，没什么思路，就先从没效率的方法开始说，对于每个数 i，检查前面从 2 到 i-1 是否存在一个数能够把它除尽。后来想想，说与其存储之前所有从 2 到 i-1 的数，还不如存这些数最后分解的因子，毕竟之前遇到的因子肯定比之前遇到的数要少得多。我再想了想，实在想不出什么好方法，就补充道应该可以有数学方法，给出什么复杂的数学公式搞定吧，不过我是学 cs 的，实在不知道额。他表示同意，说是有数学方法可以解决，不过他自己也看不懂，不要紧。这样我就放心了，然后他说那你就 coding 吧。我的 coding 虽然没什么大的 bug，但是代码有些多余的分析，他看了疑惑，觉得不够精简，因为我一开始对每个合数都不停的让之前的因子去除它，直到它不能再被除尽位置。如果剩下的值不为 1，就把剩下的值作为新的因子，加入到所遇到的因子列表中。后来发现我忽略了一个重要的问题，每个所遇到的因子，必定为之前遇到过的质数，所以这里的一个不断做除法的循环是完全没必要的。我之后精简了 code，他觉得好多了。不过我觉得我的表现他应该不是特别满意吧，毕竟一开始代码不简洁也不高效。之后他问我如果一直这样循环下去，会有什么后果。我就说 cpu 一直飞速运行，直到内存溢出，因为内存是不可能存储下所有因子的。他接着问有没有什么方法让循环持续时间更长，找到更多的质数。我说那就把这些因子分块写到硬盘上，然后每次遍历所有因子的时候按一个 block 一个 block 的来。他说能不能写下 code，我就纳闷，不就加一个 for 循环么。我直接加了个 for each 语句。他看了下，说不高

效，这样每次检查一个数是不是质数，都会遍历所有的 block，disk io 开销太大。我没听懂意思，以为说我写 for each 的时候需要预先载入所有的 block，我连忙说不是的，这些 block 是链表形式链接的，把 for each 语句改成了链表形式的遍历。他还是不满意，毕竟我没抓住重点。他说你应该把函数接口变下，输入的数据不应该是一个数，而是一个 block 的数。我恍然大悟，然后连忙接过话，说这个学过，本质就是编译器课里对嵌套循环的优化嘛，把多层循环分块，提高 data locality。然后解释说我以为他是让我在降低 time complexity 上再下功夫，没想到是让我注重算法复杂度里常数因子的开销啊。他虽然表示同意，但是感觉这个人也许是年纪大的关系，没第一个人活泼，语气很低沉，给我感觉我的表现有些糟糕。

第一个题是 10 进制转换成 26 进制，只让说怎么实现，不让实现，除法，然后取余数，然后反向输出，接着有问了如果测试，这里就有点 tricky 了，因为 26 进制，所以应该按照两位 两位的来乘以 26 来输出 原始数据， 既然让说怎么测试了，我顺便也把测 efficiency 说了一下，本来以为说完就要立马 code 了
十进制十八进制转换，十八进制加法