201 Bitwise AND of numbers range

Review II

201-1 while loop的移位条件，有时候需要看是否是最后一位了，如果是最后一位，不能进行操作以免影响结果！！！

201-2 本题的要点是 找从左往右m，n的第一个不一样的点，其后部分都归零。所以会有sol 2，先同时向右移，找到关键点后再左移回来。

202 Happy Number

ReviewII

202-1 不要猜答案，按照题目的逻辑去做。

203 Remove LinkedList Elements

204 Count Prime

Review III

204-1 请反复阅读本题答案。为什么只有在i=当前为质数时候才开启第二层循环？！为了防止重复！！

205 Isomorphic Strings

Review II

205-1 用map去比较两个string 肯定不能建立两个map：建立一个map作为两个String中字母的对应关系！

205-2 在处理1对1match的时候不只可以用containsKey 还可以用containsValue

207 Topolocial Sort for map

Review III

207-1 Tushar’s lecture about topological sort: 用set和stack可以实现topological sort.一旦有环 则做不到

207-2 方法1：建立hashMap存储dependent关系。然后遍历每个元素并且dfs她的dependent看有没有重复的元素！！！注意如果某个元素及其上游检测ok，可以标记并减少重复的查找。所以这里存储已经遍历过的元素不用hashSet，用boolean[] array 并且有-1 0 1三种可能性代表三种可能！！！

207-2 方法2： 逐个排查归类法：BFS逐渐排出每个元素！首先建立数组记录元素的dependency个数，最后结果为0的自动计入已完成的队伍。然后看谁depend on 已完成的元素，并且减去相应的值在dependency的数组中，直到该元素在数组中为0。最后看是不是所有元素都归类了

208 Implement Trie (Prefix Tree)

Review II

208-1我的解法非常好。采用randy的chaining return基本的操作单位可以连续调用，读起来非常清晰！

208-2 看到了所有的输入都是char其实可以想到用int[26] 肯定是另一种解法：所以child可以是一个TrieNode数组： TrieNode[] children = new TrieNode[26];

209 Minimum Size Subarray Sum

209-1 Slow Fast 指针对于数组中连续部分的最值

210 Topological Sort II

Review III

210-1 需要顺序的输出没有dependency的点，想到dfs，想到queue。把又有点分成两类：保证无loop的和有loop的逐渐收敛

210-2 Topoligical Sort 实际是dfs。逻辑：dfs时候，在同一loop中发现同样点，证明有loop返回false。另外就是在其他前面遍历的loop中（不是当前loop）发现的点，已经visited不用再次visited,可以直接返回true。用stack记录点，注意：只有当没有下一层的时候 再入stack，所以stack加值要在dfs的最外层

210-2 如果是单向的只是加元素，可以用hashSet。但是如果随着程序的运行存在与否经常改变，请用boolean[] array