A题

经典容斥原理

把题目拆成两个问题来看：

求[1,n]内与m互质的数的和 求[1,n]内与m互质的数的平方和

对于第一个问题 我们可以对m进行质因数分解 然后可以二进制枚举 就可以算出答案

第二个问题 本质上跟第一个问题是一样的

关于容斥原理 群里有一份容斥原理的PPT，可以参考那个

参考代码：<https://paste.ubuntu.com/p/dXNDtzPMZg/>

B题

裸的区间dp

我们令dp(I,j)表示完全消去[I,j]内的数的最小花费

那么，肯定有如下的状态转移：

根据这个式子转移即可 循环或者记忆化搜索都能通过

参考代码：https://paste.ubuntu.com/p/bVW8CJ3M45/

C题

带数量的背包问题

对于每个物品 进行二进制拆位 可以拆成若干个物品 之后用01背包的办法即可完成

整体复杂度为O(nmlogn)

背包问题可以参考崔添翼的《背包九讲》

参考代码：<https://paste.ubuntu.com/p/3ZkwhdbDky/>

D题

签到题

对每个数 直接1到sqrt(n)枚举 即可

E题

计算几何

枚举点集中的两个点

在已知半径的情况下，这两个点跟半径可以确定0个，1个或者2个圆

然后对于每个圆 判断在圆里面的点的价值的总和 取个最大的即可

注意 有可能不存在圆 所以要先找出单个权值的最大值

参考代码：<https://paste.ubuntu.com/p/s8dp5mDpKS/>

F题

dp

显然，对于链上的每个点 他们肯定不能在一个集合里面

那么，我们可以找出这个图的一个最长的链 然后这个链的每个节点分配一个集合，这样的话其他的点肯定可以分到其中的一个集合中

由于题目保证无环 所以这个图实际上就是一个dag(有向无环图)，在拓扑排序的时候进行一遍dp即可

令dp(i)表示以i为末尾的最长链长度 那么肯定有

dp(v)=max(dp(v),dp(u)+1) u为v的父节点

tips: 这个题的数据比较弱 直接dfs一遍是可以通过的，但是dfs会有很多无用的状态，在数据正常的情况下这样的做法会TLE

参考代码：<https://paste.ubuntu.com/p/BS6v7pTp4M/>

G题

三遍BFS即可

第一遍：从随便一个节点开始BFS 找到离这个节点最远的某个节点，记做U1

第二遍：从U1开始进行一遍BFS 如果存在多个距离U1最远的节点 就输出最远距离

第三遍：如果只有一个距离U1最远的节点，记做U2，那么从U2开始BFS一遍，如果有多个距离U2最远的节点，那么输出这个最远距离，否则输出最远距离-1

至于为什么需要三遍bfs 而不是两遍就能出结果，可以参考这个样例：

6

1 2

1 4

2 5

4 3

4 6

如果只进行两遍BFS的话，从1节点开始BFS，那么你的U1就会定为3节点，从U1出发，最远的节点只有1个5节点，如果没有第三遍BFS，答案就会是3

参考代码：https://paste.ubuntu.com/p/NMbwqrWQSb/