IOI2013**中国国家集训队作业CODEFORCES题目泛做**

(截止日期：第一部分11月10日，第二部分冬令营前10天，具体待定)

教练：胡伟栋唐文斌

提交方式: 通过tsinsen.com提交，将开启提交窗口

本次作业包括试题准备和泛做两部分。每位选手原则上应完成全部作业。

**第一部分：试题准备**

这一部分的截止日期为11月10日，未能按时完成的根据完成时间酌情扣分，扣分上限为全部作业分数。

作业内容：每个准备翻译两道CODEFORCES的试题，每人需要翻译的试题见下表中，将翻译好的试题添加到清橙评测系统中。试题的编辑方法见<http://tsinsen.com/help/newproblem.page>。要求：

1. 按试题的原本意思准确翻译，语言优美；
2. 试题标题使用原试题标题；
3. 试题关键字写试题所考查的算法；
4. 试题来源写：CODEFORCES ???，其中???表示题号，如CODEFORCES 240F；
5. 题目包含完整的问题描述、输入格式、输出格式、样例输入、样例输出、数据规模和约定等部分，使用清橙的格式化后试题可正常阅读；
6. 至少20个数据，数据有梯度、有区分度，包含至少两个可以手算的数据；
7. 提交好正确的参与程序，此参考程序必须在CODEFORCES上测试通过；
8. 试题设置为共享。

**第二部分：试题泛做**

这一部分的截止日期为冬令营前10天。

作业内容：每人做完表格中所有试题，提交到CODEFORCES上通过，并在清橙评测系统上提交通过，并填写表格中的题目大意（无需背景，说明模型即可）、算法讨论/说明、通过情况。在清橙作业中提交最终表格。

**[泛做表格] (完成情况: \_\_ /100)**

**CODEFORCES用户ID: \_\_csimstu\_\_\_\_**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **编号** | **负责人** | **题目名称** | **题目大意** | **算法讨论** | **Status** |
| 7E | 贾志鹏 | Defining Macros | 给定n个#define name value的语句，其中value包括+-\*/()，判断另一个语句是否会产生歧义。 | 每个变量设4种状态：有歧义(S)，多段相加减(P)，多段相乘除(M)，一段(Q)。然后分情况讨论：  A+B：不影响；  A-B：B不能是P；  A\*B: A,B均不能是P;  A/B: A不能是P，B不能是P,Q；  用递归可以简化编程复杂度。 | Accepted |
| 8D | 贾志鹏 | Two Friends | 平面上三个点H,S,C。两人同时从C出发，路线分别为C->H和C->S->H。两人均有一个路程的上限值。求两人最长公共路线。 | 如果经过shop，那么答案就是min(t1,t2)，只要可以走到。  否则，二分答案x，那么两人同行的终点一定在半径为x的圆内，设为p。  同理，p也要在以home,shop为圆心的两个圆内。于是变成了判断3个圆是否有交，  枚举两个圆算交点、各个圆的圆心，看是否满足。 | Accepted |
| 8E | 王康宁 | Beads | 将1-2^(n-1)的数分组，同一组的数可以通过逆序和01取反得到，然后每组保留最小的数。求第k大的数。 | 先二分答案，然后计算<=k的有多少个满足条件的数：  原问题对x所属组的定义是{x,~x,inv(x),inv(~x)}。定义子问题1中的组：{x,inv(x)}，子问题2的组：{x,inv(~x)}，然后分别解决以上问题：  原问题：  1. 0XXX0 -->子问题1  2. 0XXX1 -->子问题2  3. 1XXX0 -->impossible  4. 1XXX1 -->impossible  子问题1：  1. 0XXX0 -->子问题1  2. 0XXX1 -->全部合法  3. 1XXX0 -->impossible  4. 1XXX1 -->子问题1  子问题2：  1. 0XXX0 -->全部合法  2. 0XXX1 -->子问题2  3. 1XXX0 -->子问题2  4. 1XXX1 -->impossible  于是可以设计3个相似的dp，每个dp解决不超过二分的答案的合法数的个数。 | Accepted |
| 10E | 王康宁 | Greedy Change | 给定一个货币系统，求最小的s，使得贪心（从高价值货币开始能选就选）使用的货币不是最优解。 | 论文题。 | Accepted |
| 15E | 陈立杰 | Triangles | http://codeforces.ru/renderer/5945210be972aa5fe947f3b8a3a0378a4cade844.png求从H出发再回到H的路径数，满足不自交，且形成的封闭区域不含阴影三角形。 | 显然一条满足要求的路径必然是左边走一圈，回到最靠上阴影三角形的上定点，然后再在右边走一圈。仔细观察可以发现很多递归子问题，一个是往中间走然后出来，一个是往下走然后回去。设前者为f[i],后者是g[i]，不难发现递推式：  f[1]=1;f[i]=2f[i-1]+3;  g[n/2]=2;g[i]=2+2f[i]+f[i]\*g[i+1];  答案就是2(g[1]^2+1)。 | Accepted |
| 17E | 陈立杰 | Palisection | 给你一个长度 n (1 <= n <=2\*10^6) 的只由小写字母组成的字符串s，问s的所有连续回文子串中，有几对是有公共部分的。 | 先用manachan算法求出每个位置的最大回文半径。直接统计讨论比较麻烦，可以采用补集转化思想，求不相交的对数。对每个位置求出f1[i]表示以位置i开始的回文段数，f2[i]表示以位置i结束的XXX。所以不合法的答案可以由f1,f2拼凑。 | Accepted |
| 17C | 余翔 | Balance | 给定一个长度为n(1<=n<=150)的abc字符串，每次可以将相邻两个字符c1,c2进行以下变换：c1c2=c1c1或c1c2=c2c2。求进行任意次变换后可能的字符串总数。 | 可以发现，原字符串中序是不变的。故可设计以下状态：f[i][j][na][nb]表示构造到第i个字符，目前考察原字符串第j个字符，a,b的数目。转移枚举下一个字符，为了避免重复，强制下一个字符由原串中j之后最近的来产生。总复杂度为O(n^4) | Accepted |
| 19E | 余翔 | Fairy | 对于一个n个点m条边的无向图（n,m<=10^4），问删除哪条边后变成二分图。 | 首先建一个生成树森林，然后进行染色。对于每条非树边，如果两端点的颜色不同，则称为errEdge。在树上连接errEdge两端点的路径称为errPath。首先考虑删除非树边：如果n(errEdge)=0那么都可以删除；如果n(errEdge)=1则必须删除这条errEdge；如果n(errEdge)>1那么都不能删除。然后考虑树边：对每条树边记ai,bi分别表示经过这条边的errPath数目，非errPath数目。于是可以发现，第i条树边是答案的充要条件是n(errEdge)=0 or (ai=n(errEdge) and bi=0)。用各种数据结构都可以维护。 | Accepted |
| 23D | 杨志灿 | Tetragon | 一个有3个等边的凸四边形，告诉你3条等边的中点，求这个凸四边形。 | 一个想法是三分半径，计算圆交点，看是否过直径。但数据组数太多，会TLE。  另一个方法是枚举中间那个点p1,那么有另外两个四边形上的点在p1与p2,p2与p3的中垂线上。解2元1次方程即可。 | Accepted |
| 23E | 杨志灿 | Tree | 给定一个树，可以删除任意数目的边，使各个连通块大小的积最大。 | 可以证明，任何一个最优解不存在一条长度>=3的路径。否则，将中间那条边割开，设形成的两部分大小为a,b(a,b>=2)，于是1>=1/a+1/b 🡪 a\*b>=a+b。有了这个原则，一个点只有三种情况：   * 这个点与它一个儿子以及那个儿子的儿子同属一个连通块 * 这个点与它多个儿子同属一个连通块 * 这个点与它的父亲同属一个连通块   这可以用经典的树形dp+背包解决。另外，本题需要使用高精度。 | Accepted |
| 26E | 向鹏达 | Multithreading | 有N(N<=100)个子过程：  repeat *ni* times  *yi* := *y*  *y* := *yi* + 1 end repeat  其中y为全局变量，而yi则相互独立。每次可以命令一个子程序运行当前行并跳到下一行。设计一个长度为2\*N的操作序列，使得最后y = W(|W|<=10^9)。(ni<=1000) | 构造题。首先，如果不满足1<=W<=sigma(ni)一定无解。否则有以下三种有解情况：   1. N=1, W=n1。这种情况很显然。 2. N>=2,W>=2。可以设计一个W=2的序列：1,(3到n全部用掉),(n2-1)次2,1,2,(n1-1)次1,2。然后对于多出来的W-2，将中间的循环移到最后即可。 3. N>=2, W=1且存在ni=1: i, everything else, i. | Accepted |
| 28D | 向鹏达 | Do not fear, DravDe is kind | 有n个人按1-n从左到右排成一排。每个人有四个属性(v,c,l,r)。第i个人要求他左边人的c之和恰好为l,右边人的c之和恰好为r。现在要剔除一部分人，使每个人的需求得到满足，且v之和最大。 | 如果选定一个人k，那么可以确定最终队列中所有人的c之和必为(ci+li+ri)。故可以按每个人的c+l+r分组，不同组之间互不干涉，这样一来限制就只有l了。于是转化成了背包模型。不同的是转移总数只有O(n)次，有用的状态很少。于是可以对状态进行hash再按常规的背包dp来做。 | Accepted |
| 30D | 罗雨屏 | Kings Problem? | 有n+1个平面上的点，前n个都在x轴上。现在求从k号点出发，经过每个点至少一次的最短路径。 | 如果当前在n+1号点，那么之后一定是往还没访问过的最左或最右走，再到达另一极端。于是可以枚举经过n+1之前的访问区间[l,r]。可以发现，[l,r]必定包含最左端点或最右端点才能最优，故可用区间只有O(n)个，而验证是O(1)的。 | Accepted |
| 30E | 罗雨屏 | Tricky and Clever Password | 给定一个加密后的奇数长度的回文串：A+prefix+B+middle+C+suffix，其中prefix,suffix为原串的首尾等长的子串，middle为长度是奇数的回文串。求一种破解方法，使prefix+middle+suffix，即原串，的长度最大。 | 首先预处理出每个位置的最大回文半径。可以证明，若middle以某个位置为中心，那么当middle恰好为以那个位置为中心的最长回文串，得到的解不会更差。再KMP预处理另一个数组f[i]表示与字符串[i,n-1]（n为长度）逆序匹配的最靠前位置。于是可以枚举每个middle，注意到当i增加时，f[i]不增，所以可以二分位置k使f[k]在middle之前且k最靠左。整个算法的复杂度为O(nlogn)。 | Accepted |
| 32E | 乔明达 | Hide-and-Seek | 平面上有一堵墙和一面镜子，以及两个可以视作点的人。问两人是否可以看到对方。只要有公共点即视作相交。 | 最简单的情况：如果两人连线没有被挡着，那么可见；否则，如果可行一定是通过了镜子，如果镜子挡到视线那么不能看到对方。通过对称可以确定镜子上一个必然经过的点，判断这个点与两人的初始位置是否无障碍即可。 | Accepted |
| 35E | 乔明达 | Parade | 有n个房子坐落在x轴上，每个房子可以视作有3个属性的矩形：l,r,h，分别表示房子的左、右边界以及高度。求一个封闭区域面积最小的轮廓，将所有房子包含在内。 | 离散化端点，然后求出每相邻两个离散点的最大高度，扫一遍得到答案。 | Accepted |
| 36E | 朱新瑞 | Two paths | 给定一个无向图，求两条欧拉路径将所有边无重复地覆盖。 | 1. 连通块数>=3，无解 2. 连通块数=2，在两个连通块内分别求欧拉路径。 3. 连通块数=1，如果有0个或2个奇度数的点，那么存在一条欧拉路径，任意将之分成两段即可；如果有4个奇度数的点，那么任选两个添加一条辅助边，求出欧拉路径后以这条边为分割即可求出答案。 | Accepted |
| 37E | 朱新瑞 | Trial for Chief | 给定一个n\*m(n,m<=50)的黑白棋盘，每次可以将一个四连通块反色。求最小操作次数使整个棋盘变为白色。 | 不难发现，最优方案一定是不断反色某个格子所在连通块。于是枚举每个格子，然后bfs求出到边缘的最长黑白交错路径即可。 | Accepted |
| 39C | 罗干 | Moon Craters | n条x轴上的线段{(li,ri)|1<=i<=n}，要求选出一个集合，使不存在两条线段相交（可以互相包含），求使集合最大的方案。 | 先离散化坐标，设计状态f[i][j]表示[i,j]区间的答案。枚举以j为尾的线段，那么f[i][j]<-max(f[i][j-1],f[i][seg.l]+f[seg.l][r])。如果[i,j]本身是一条线段，那么f[i][j]+=1。转移时记录方案，最后递归输出答案。 | Accepted |
| 39A | 罗干 | C\*++ Calculations | 定义一种如下的语言：   * *expression* ::= *summand* | *expression* + *summand* | *expression* - *summand* * *summand* ::= *increment* | *coefficient*\**increment* * *increment* ::= a++ | ++a * *coefficient* ::= 0|1|2|...|1000   其中，a的初始值已知。求一种运算顺序，使给定表达式的值最大。 | 将表达式解析成很多k\*(++a)或k\*(a++)的形式。显然，因按k升序来操作。而k\*a+k\*(a+2)=2\*k\*(a+1)，故++a和a++的顺序不影响。 | Accepted |
| 39E | 龙浩民 | What Has Dirichlet Got to Do with That? | 一个游戏，初始时a=a0,b=b0。两人轮流操作，每次可以将a+=1或b+=1，如果使a^b>=n，则输。问谁有必胜策略。(a<=10^4,b<=30,n<=10^9) | 如果a=1，那么如果b>=log2n则平局；  如果b=1，那么当a>=sqrt(b)则N，P状态交替出现；  否则递推就行了。表的大小只有O(log2n \* sqrt(n))。 | Accepted |
| 40E | 龙浩民 | Number Table | 给定一个k个各之填有1和-1的n\*m棋盘，保证k<max(n,m)。一种合法方案指将剩下所有各子填满1,-1后满足每行每列乘积为-1。求方案数。 | 如果n,m奇偶性不同，则无解。否则不妨设n>=m。通过交换行可以使最后一行为空。注意到只要前n-1行填完，那么最后一行自然确定。所以自由格子数就是每行未确定格子数-1的和s。答案就是2^s。 | Accepted |
| 43E | 谭震 | Race | 在一场赛车比赛中，有n(n<=100)辆赛车。每辆赛车有k(k<=100)段行程，第i段行程的速度为vi，时间为ti。求总共的超车次数。 | 枚举两个赛车，然后O(k1+k2)统计答案。方法是维护两个指针。情况有：   * 两线段交点处超车🡪线段相交判断 * 在两条线段的公共端点上超车🡪特判 * 在某条线段的端点上超车，另一条线段经过该点🡪特判 | Accepted |
| 44J | 谭震 | Triminoes | 给你n\*m(n,m<=1000)的棋盘，其中每个格子可以是黑白两色，亦可以没有颜色。现在要用1\*3的骨牌覆盖它，要求骨牌中间要对应黑色，两端对应白色。求一种合法方案，或者判断无解。其中合法方案要求用4种颜色去染，相邻骨牌颜色不能相同。 | 首先，如果n(黑色)!=n(白色)/2，那么无解；否则，如果可以唯一确定的全部加入一个队列。然后每次处理完队列的一个元素可能会导致相邻原来不是唯一确定的变成唯一确定，于是统统入队。显然，这个算法一定会停止，因为不可能所有都是无法确定的。对于4染色，暴力就可以了，即每次选相邻未出现的颜色染。貌似没有反例。 | Accepted |
| 45G | 毕克 | Prime Problem | 将1-n(n<=6000)的整数分成尽量少的份数，使每份的和是质数。 | 由哥德巴赫猜想，任何一个>2的偶数都可以拆成2个质数的和；任何一个>7的质数都可以拆成3个质数的和。所以答案不会超过3。拆成两个的情况比较简单，三个的话单独将3分为一组划归成两个的情况。 | Accepted |
| 45E | 毕克 | Director | 有n(n<=100)个name，n个surname。现在要将name和surname配成n对，然后排成一排。要求首字母相同的对数尽量多。其次，最后排成一排的字典序最小。 | 将name和surname单独排序。从小到大枚举每个name，考虑跟surname匹配。如果匹配后不会影响首字母相同的总对数，就贪心匹配最小的surname。 | Accepted |
| 46F | 黎文杰 | HerculePoirot Problem | 给定一个犯罪前和犯罪后的场景。场景中有n个房间，m扇门，k个人。一共有m把钥匙，分别对应每一扇门。初始时所有房间都上了锁。一个人可以打开一扇门只要他有对应的钥匙。同一个房间的人之间可以交换钥匙。一个人可以走到相邻的没有上锁的房间内。以上3个动作可以发生任意次。要求判断两个场景是否吻合。 | 注意到，人的每种行动都是可逆的。也就是说，只要两个场景可以通过某种方式到达一个完全相同的状态，那么这两个场景就是吻合的。如果两个房间可以通行，那么称它们在一个连通块中。于是可以尽可能的膨胀每个连通块，最后直接比较房间、钥匙、人所属的极大连通块是否一样。这可以用并查集来实现。 | Accepted |
| 47E | 黎文杰 | Cannon | 有n(n<=10^4)个导弹，m堵墙(m<=10^5)。导弹从(0,0)发射，初速度均为V，发射角度各异，但都<=45°。墙是指(xi,0)-(xi,yi)的垂直线段。问每个导弹最终坐标。 | 将导弹按角度排序，墙按x坐标排序。如果第i枚导弹可以越过第j个墙，那么i之后的导弹也可以。用两个指针扫一遍即可。 | Accepted |
| 49E | 王若松 | Common ancestor | 有两个DNA序列s1,s2（length<=50），求最短公共祖先DNA序列。DNA序列由26个小写字母组成。每次衍生可以看做是c1->c2c3，其中c1是原来某个字符，c2c3是衍生出来的字符，且在新的DNA序列中相邻。一个串t的祖先串p指通过一系列的衍生p可以形成t。 | 预处理出can[c][l][r]，表示s1（或s2）的[l,r]子串是否可以由字符c衍生出来。再进行dp:f[i][j]表示形成s1[1..i]与s2[1..j]的最短公共祖先DNA序列的长度。转移：f[i][j]🡨min(f[ii][jj]+1)，如果s1[ii+1..i]与s2[jj+1..j]可以由某个字符衍生得到。 | Accepted |
| 51F | 王若松 | Caterpillar | 毛毛虫是指一棵树中存在一条路径，使得所有点到这条路径的距离不超过1。给定一个无向图，每次可以合并两个点。问最少需要多少次操作可以将其变为毛毛虫。 | 显然，对于边二连通分量中的k点，必定要合并k-1次来缩成一个点。这样一来，就变成了一片森林。如果两个树都是毛毛虫，那么可以一定通过某种方法一次操作合并成一个毛毛虫。反之，如果有一个不是毛毛虫，那么合并后无论如何也不是毛毛虫。于是原问题简化成了一棵树的问题。另一个发现是，毛毛虫的主轴一定是连接两个叶子节点。如果不是可以继续延长至叶子，同样也是毛毛虫。所以枚举每个叶子，以它为根做一次dfs。一边遍历一边统计更新答案。复杂度是O(n^2)的。 | Accepted |
| 53E | 王启圣 | Dead Ends | 给定一个n(n<=10)个点的无向连通图，删除一些边后使其变成一棵树。要求最后叶子节点个数是k。求最后树形态总数。 | 很明显，添加边比删边要容易得多。由于只有n个点，可以考虑强行记录所有状态。为了防止重复统计，强制定1号节点为根。用f[s1][s2]表示已用节点集合为s1，叶子节点集合为s2，其中s2∈s1。对于一个有t个叶子的状态，考虑这个状态由哪些状态转移过来。可以发现，恰好被重复统计t次！于是就迎刃而解了。还有一个问题，就是当1号节点为叶子时没有统计到。可以多开一维状态来记。 | Accepted |
| 57D | 王启圣 | Journey | 给定一个n\*m(n,m<=1000)的棋盘，其中有些格子不能经过。求从任意一点（非障碍）出发，到任意一点（非障碍）结束的最短路期望长度。 | 首先可以O(n^2)求出任意两点的曼哈顿距离总和。注意到最短路要么等于曼哈顿路，要么等于曼哈顿路长度+2。  \*X...............  \*\*\*\*X............  \*\*\*\*\*\*\*\*\*X.......  \*\*\*\*\*\*\*\*\*\*\*X?????  如上图所示，从\*到?的最短路需要多加2。而\*成不间断的阶梯型。往下是对称的故不讨论。  还有一种是竖着的类似情况，也完全一样地处理。  注意到后面其实是O(n)的。实际上如果前面求和时把绝对值符号拆开讨论，整个算法也能做到O(n)。 | Accepted |
| 226E | 汤沛雯 | Noble Knight's Path | 给定一棵n(n<=10^5)个点的树，每个点有一个权值v(v<=10^5)，同时有q(q<=10^5)个询问，每个i询问有向路径(ai,bi)上满足v不属于[li,ri]的第ki个点。 | 典型的可持久化线段树。对每个点建一个它到根以权值为关键字的线段树。然后预处理倍增数组anc。对于每个询问先看是否有解。如果有，确定是在由lca分隔开的哪半条路径上。然后利用倍增数组贪心不断往上跳，直到不能跳为止，答案就是那个点。判断能不能从u跳到p就要用到可持久化线段树，询问u到根路径上满足条件的点个数和，再减去p的父亲到根路径上的个数和。 | Accepted |
| 217D | 汤沛雯 | Bitonix' Patrol | 给出n(n<=10000)个常数a1...an，求去掉一些常数后，使sigma(xi\*ai)=0(mod m),xi∈{-1,0,1}无解的方案数。(m<=120) | 显然，如果按ai mod m分类，那么同类至多只能选1个；由抽屉原理，最多只能选6个数，否则>=2^7>=120。于是只有C(60,6)≈5000W种情况。爆搜即可。要用位运算加速。 | Accepted |
| 67E | 钱雨杰 | Save the City! | 给定一个简单多边形，其中一条边AB平行于x轴。问AB上有多少整点可以看到所有所有端点。 | 对每个端点找出AB上对应可以看到的线段，最后求交即可。用向量叉积可以分4种情况讨论。 | Accepted |
| 67C | 钱雨杰 | Sequence of Balls | 给定两个序列A,B（长度<=4000），通过以下四种代价不同的操作将A变成B：   1. 删除任意一个字符 2. 添加任意一个字符 3. 交换相邻两个字符 4. 将任意一个字符修改为另一个字符。   其中，保证2\*3操作的代价>=1,2操作的代价。 | 对于“保证2\*3操作的代价>=1,2操作的代价”，意味着一个字符不可能交换多次。设计状态f[i][j]表示A的前i个跟B的前j个字符匹配下的答案。1,2,4都比较方便实现。交换比较麻烦：  A:a\*\*\*b  B:b......a  那么可以将\*全部删除，再插入.。 | Accepted |
| 70D | 王迪 | Professors task | 动态凸包，支持插点和询问一个点是否在凸包内。 | 建立循环的极角序，用set维护。 | Accepted |
| 70E | 王迪 | Information Reform | 给定一个n（n<=180）个节点的树，需要在其中选择一些点放置基站，每个基站代价为k。其余的点到距离自己t的基站的通讯费用为d[t]。求一种代价最小的方案。保证d[i]<=d[i+1]。 | 这个题的dp不是很好想。首先任意定根。用f1[i]表示以i为子树的答案，f2[i][j][k]表示在i子树中处理完j子树，且i选择k作为基站的答案。转移根据j子树是独立出来还是选k为基站讨论。 | Accepted |
| 156E | 高胜寒 | Mrs. Hudson's Pancakes | 有n个调料，编号为0-n-1。还有m道食谱，每个食谱有3个属性d,s,c。称一个调料能匹配一个食谱，意味着d进制下通过修改s中的?可以得到食谱的编号。一个食谱的价值为它能匹配的调料编号的积+c。求每个食谱价值的最小质因数。 | 首先，预处理出所有f[d,s]=mul(i|i属于s)。s不会太多，因为如果位数大了，前面必为0。又因为100以内质数只有25个，可以再开一维记模的答案。然后就是各种常数优化，比如每5个质数分成一组来记模。 | Accepted |
| 105D | 高胜寒 | Entertaining Geodetics | 有一个n\*m(n,m<=300)的棋盘，每个格子有一个颜色。某些格子上放有某个颜色标记，同时有一个标记队列。初始时将某个位置的标记入队。然后不断重复以下操作，直到队列为空：  取出队首标记。如果该标记所在格子颜色不为0且标记与格子不同色，那么所有该格子颜色的格子会按螺旋形重新染成该标记的颜色。如果在重染色的时候格子上有标记，那么将其入队。问总共染色次数。 | 题意异常纠结。可以注意到，一旦进行一次螺旋形染色，那么某种颜色的格子上所有的标记将全部入队。并且，如果某个格子之前被重染色过，那么之后每次染色都会受到影响。记下来统计答案即可。 | Accepted |
| 193D | 雷凯翔 | Two Segments | 给出1-n的一个排列p，求不同的(a,b,c,d)的个数，满足a<=b<c<=d且{p[a],p[a+1],...,p[b]}∪{p[c],p[c+1],...,p[d]}得到的集合是公差为1的等差序列。判断两个四元组是否算重，看生成的集合是否一样。n<=3\*10^5 | 定义一个函数f(a,b)，表示形成以a为首项b为末项的数在原序列中的段数。那么答案就是sigma([f(a,b)=2]+[f(a,b)=1])-n。如果将(a,b)->f(a,b)建成一张表，可以发现以下性质：   1. 每个元素和它相邻的元素差不超过1。特别的，每个元素和它下面的元素差不超过1。 2. 如果将每行减去它下面一行，那么得到的是如下形式的序列：1,1,...,1,0,0,...,0,-1,-1,...-1，是随右端点增大而单调不增的。   基于这两个性质，可以从下往上用线段树实现区间加、减，询问最小、次小值（因为全部为正数，1,2必然是最小的两个）来统计答案。 | Accepted |
| 75E | 雷凯翔 | Ships Shortest Path | 在一片海域中，有一座孤岛（凸多边形）。一艘船要从A到B，要求航程最短，且只能经过岛上或AB连线。可以经过岛上，但航程要按双倍记。 | 求出交点，然后分情况讨论。为了避免边界精度问题，可以最后用floyd简化。 | Accepted |
| 76F | 郭志芃 | Tourist | 在x轴上，有n(n<=10^5)个事件点。每个事件点有坐标和发生时刻。一个人从某个点出发，两个方向都可以选择，速度不超过V。求一种方案使观看到的时间个数最多。 | 将每一个事件点(ti,xi)映射到(xi,yi)。如果可以从i走到j，那么|yi-yj| <= |xi-xj|\*V，也就是说I,j两点的连线斜率绝对值不超过V。于是变成了经典的二维最长不下降子序列问题。可以O(nlogn)解决。 | Accepted |
| 76A | 郭志芃 | Gift | 给定n点(n<=200)，m(m<=5\*10^4)条边的无向图。每条边i有两个权值ai,bi。求确定两个参数S, G，使得所有a<=S,b<=G的边连通。使S\*kS+G\*kG最小。 | 将所有边按a排序，然后依次处理。每次添加一条边，置换出那个环上b值最大的边。然后每次暴力统计答案。总复杂度O(NM)。 | Accepted |
| 77E | 罗剑桥 | Martian Food | 一开始有两个内切的圆，半径为R,r(R>r)。然后开始在大圆内添加新的圆，使其与初始时的两个圆一个内切一个外切，且面积最大。问这样下去第k(k>=10^4)个圆的半径。多组数据(<=10^4)。 | 以切点为原点，过两圆的直径为x轴建立极坐标系。对每个点(d,φ)，进行变换(d,φ)<->(1/d,φ)。于是过原点的两个圆被变换成两条垂直于x轴的直线，而其他圆依然变换成圆，添加进去的圆夹在两直线之间。于是，在新坐标系中第k个圆心为((R+r)/4Rr,k(R-r)/2Rr)。注意，这个圆心不是对应原坐标系中的圆心。应该取三个圆周上的点，反推回去求三角形外接圆半径。复杂度是常数级别。 | Accepted |
| 79D | 罗剑桥 | Password | 一排有n(<=10^4)个01格子，初始时全为0。给出一个目标状态，保证1的个数不超过10，以及一些操作，不超过100个。每个操作i由ai确定，意味着可以将长度为ai的连续区间取反。每个操作可以进行任意次。求最少总操作次数。 | 首先，需要简化操作的格子数。有两种方法：前缀xor或者相邻两项xor，而后者可以保证变换后1的个数依然不超过20，故采用后者，得到一个新数组b。于是每次操作变为将距离为ai的两个格子取反。注意到可以将操作看作是从一个格子跳到另一个格子，于是用最短路可以求出将任意两个1格子取反的最少操作数。最后状压dp即可。 | Accepted |
| 81E | 郑舒冉 | Pairs | 给出一个章鱼型图（每个点有且仅有一条出边），每个边有双关键字的权值。求选出一些边使得没有一个点同属于多个边，且权值和最大。点数<=10^5。要求输出方案。 | 这题思路还是比较显然的。定义一个关键字类，重载运算符后当一个关键字处理。先在树上dp，再在环上dp。输出方案按逆dp序来。 | Accepted |
| 82E | 郑舒冉 | Corridor | 在平面中，y=h和y=-h是两堵墙。在墙上面有一些窗户，用线段(l,h)-(r,h)表示，且关于y=0对称。在(0,f)和(0,-f)有两个光源。求两堵墙之间被照亮的面积。 | 注意到每个点不可能被覆盖超过2次。可以考虑用简单容斥把覆盖了2次的面积减掉。枚举y=h上和y=-h上两两窗户，求半平面交即得答案。 | Accepted |
| 83E | 孙伟峻 | Two Subsequences | 定义一个函数f：   * *f*(empty sequence) = empty string * *f*(*s*) = *s*. * *f*(*s*1, *s*2) =  the smallest in length string, which has one of the prefixes equal to *s*1 and one of the suffixes equal to *s*2. * *f*(*a*1, *a*2, ..., *an*) = *f*(*f*(*a*1, *a*2, *an*- 1), *an*).   给定等长字符串序列a1...an，需要将其分成两个子序列s1,s2，使得|f(s1)|+|f(s2)|最小。n<=10^5，字符串为01串，且长度<=20。 | 考虑一个暴力的dp：f[i][j]表示到了第i个字符串，另一个序列以第j个字符串结尾。转移枚举第i+1个字符串是接到哪个序列上去。尽管转移时O(1)，但光状态就是O(n^2)。于是想办法简化状态。可以设f[i]表示i与i+1属于不同序列的答案。那么转移就需要枚举一段区间：f[i]=min{f[j]+w[j+1][i]-g(j,i+1)}，其中w[l][r]表示将l到r的字符串压缩后的长度，g(i,j)表示将第i和第j个字符串接起来节省的长度。进一步观察可以发现，g(j,i+1)的取值只有20种。故可以枚举j与i+1的公共长度，用位运算来记录后缀为某个串的答案。于是整个算法复杂度只有O(n\*20+2^20)。 | Accepted |
| 85E | 孙伟峻 | Guard Towers | 平面上有n(<=5000)个点，可以将其染成黑色或者白色。求一个p的最小值，使最远黑点距离<=p且最远白点距离<=p。坐标范围<=5000。距离指曼哈顿距离。 | 算出两两之间的曼哈顿距离，然计数排序。从大到小枚举曼哈顿距离（不超过10000），看答案能否为当前枚举的值。判断方法是动态维护一个二分图，支持动态加边。这可以用并查集暴力O(n)合并实现。总复杂度为O(n^2)，常数稍大。 | Accepted |
| 86E | 王子昱 | Long sequence |  |  |  |
| 89D | 王子昱 | Space mines |  |  |  |
| 91D | 国家琦 | Grocers Problem |  |  |  |
| 93D | 国家琦 | Flags |  |  |  |
| 97C | 杜瑜皓 | Winning Strategy |  |  |  |
| 97A | 杜瑜皓 | Domino |  |  |  |
| 98D | 魏鑫鼎 | Help Monks |  |  |  |
| 98C | 魏鑫鼎 | Help Greg the Dwarf |  |  |  |
| 191D | 孙猛 | Metro Scheme |  |  |  |
| 164D | 孙猛 | Minimum Diameter |  |  |  |
| 150E | 温和 | Freezing with Style |  |  |  |
| 101E | 温和 | Candies and Stones |  |  |  |
| 103E | 胡渊鸣 | Buying Sets |  |  |  |
| 105E | 胡渊鸣 | Lift and Throw |  |  |  |
| 107D | 刘定峰 | Crime Management |  |  |  |
| 113D | 刘定峰 | Museum |  |  |  |
| 115D | 高远 | Unambiguous Arithmetic Expression |  |  |  |
| 120I | 高远 | Luck is in Numbers |  |  |  |
| 123E | 陈文潇 | Maze |  |  |  |
| 125E | 陈文潇 | MST Company |  |  |  |
| 193E | 彭天翼 | Fibonacci Number |  |  |  |
| 145D | 彭天翼 | Lucky Pair |  |  |  |
| 132E | 许昊然 | Bits of merry old England |  |  |  |
| 138D | 许昊然 | World of Darkraft |  |  |  |
| 140F | 钟泽轩 | New Year Snowflake |  |  |  |
| 147B | 钟泽轩 | Smile House |  |  |  |
| 152D | 成羽丰 | Frames |  |  |  |
| 183D | 成羽丰 | T-shirt |  |  |  |
| 217E | 李煜东 | Alien DNA |  |  |  |
| 135E | 李煜东 | Weak Subsequence |  |  |  |
| 163D | 黄嘉泰 | Large Refrigerator |  |  |  |
| 167E | 黄嘉泰 | Wizards and Bets |  |  |  |
| 232D | 李凌霄 | Fence | 给定一个序列a1,a2,…,an,(n <= 10^5)以及q个询问(l1,r1),(l2,r2),…,(lq,rq)。对于每个询问(li,ri)，回答(lj,rj)二元组的个数，满足rj-lj+1=ri-li+1，且[lj,rj]∩[li,ri]=空，且对于所有0<=k<=ri-li，使a[li+k]+a[lj+k]=a[li]+a[lj]。 | 先将原序列相邻两项做差得到长度为n-1的b序列，就变成了询问一段取反后相同的且不交的段数。构造一个s序列为b$-b，然后就变成了经典的后缀数组离线询问的问题了。复杂度为O(nlogn+qlogn) | Accepted |
| 175E | 李凌霄 | Power Defence | 在一个塔防游戏中，boss从x轴的负无穷走到正无穷，可以在y=1和y=-1两条直线上的整点处建三种塔：火、电、冰。每种塔有一个攻击半径和每秒伤害（除冰塔外）。在某一时刻，boss走到k个冰塔的攻击半径内，则移动速度减为原来的1/(k+1)。给定每种塔的总数(<=20)，求最大化伤害的方案。 | 首先，可以吧y=-1的塔对称到y=1上，问题变成了在y=1上每个整点可以修不超过2个塔。可以发现，塔必然是连续的一段，且不能有相邻位置都只有1个塔,所以可以暴力出冰塔的修建方案。之后进行dp:用f[i][j][k]表示在前i个位置修了j个火，k个电塔，O(1)转移。优化：所有塔肯定必须用，塔连续一段的长度<=13（实际上10就够了）。 | Accepted |
| 176D | 吕可凡 | Hyper String |  |  |  |
| 178F2 | 吕可凡 | Representative Sampling (30 points) |  |  |  |
| 178E3 | 何琦 | The Beaver's Problem - 2 (50 points) |  |  |  |
| 180B | 何琦 | Divisibility Rules |  |  |  |
| 185D | 李凌劼 | Visit of the Great |  |  |  |
| 187D | 李凌劼 | BRT Contract |  |  |  |
| 176E | 陈睿 | Archaeology |  |  |  |
| 196D | 陈睿 | The Next Good String |  |  |  |
| 198E | 钱迪晨 | Gripping Story |  |  |  |
| 200E | 钱迪晨 | Tractor College |  |  |  |
| 200A | 李宇轩 | Cinema |  |  |  |
| 201E | 李宇轩 | Thoroughly Bureaucratic Organization |  |  |  |
| 201D | 孟凡航 | Brand New Problem |  |  |  |
| 204E | 孟凡航 | Little Elephant and Strings |  |  |  |
| 207B1 | 龚拓宇 | Military Trainings (20 points) |  |  |  |
| 207A2 | 龚拓宇 | Beaver's Calculator (30 points) |  |  |  |
| 167D | 张闻涛 | Wizards and Roads |  |  |  |
| 209C | 张闻涛 | Trails and Glades |  |  |  |
| 212B | 孙伟 | Polycarpus is Looking for Good Substrings |  |  |  |
| 212D | 孙伟 | Cutting a Fence |  |  |  |
| 212C | 周誉昇 | Cowboys |  |  |  |
| 213E | 周誉昇 | Two Permutations |  |  |  |
| 217C | 许瀞云 | Formurosa |  |  |  |
| 229E | 许瀞云 | Gifts |  |  |  |