试题泛做

季智成

编号	试题名称	题目大意	算法讨论	时间复杂度	空间复杂度
9293	Graph Game	给一个 n 个点，n 条边的无向连通图，并进行以下运算： 让我们定义变量 totalCost，初始 totalCost=0。然后，solve(T) (现在 T 是一个图) 1.totalCost=totalCost+(size of T).运算符'='表示赋值。(Size of T)表示图 T 中的结点个数。 2.在图 T 中随机选择一个结点 x(图 T 中每个点被选中的概率相等) 3.从图 T 中删除结点 x 4.然后 T 变成了一些联通快 5.分治处理所有的 Solve(S) (S 是剩下的连通块) 求 totalCost 的数学期望 n≤3000	将事件"A 将要被删除时与 B 连通"定义为 E（A,B）。那么每一个 E（A，B）对 totalCost 的贡献为 1。因此可以求出所有 E（A，B）的概率然后累加。 关键在于如何求 E（A，B）。首先考虑树的情况，假设有 m 个点在 A，B 这条路上，当且仅当先删除 A 时才发生，概率就是 1/n。原题是一个环加外向树，因此 A-B 之间至多有两条路。记环上一边为 X，一边为 Y，环外路径总长为 Z，那么 E（A，B）= 1/(X+Z-1) + 1/(X+Y-1) -1//(X+Y+Z-1)。	$O(n^2)$	O(n)
9279	Maxim and Increasing Subsequence	一个长为 n 的序列，重复 t 次，求最长上升子序列长度。有 k 个长为 n 的序列，分别求解。序列中的数不超过 maxb k≤10 $n≤10^5$ $t≤10^9$ $maxb≤10^5$ $n*maxb≤2*10^7$	首先可以看出，过大的 t 是没有意义的。很容易发现，t<min（n,maxb）。 定义 dp[i][j]表示到第 i 个位置时数的大小等于 j 的最后一个数位结尾的最长上升子序列长度。再用 f[j] 记下当前不超过 j 的 dp[i][k]。	O(n*maxb*T)	O(maxb)

编号	试题名称	题目大意	算法讨论	时间复杂度	空间复杂度
9280	Evil	在一个二维平面上有 n 个城市。两个城市之间的距离等于它们之间的曼哈顿距离。求出给定城市的最长可能的哈密尔顿回路的长度。 $3\le n\le 10^5$，$0\le x_i,y_i\le 10^9$	对 x，y 坐标排序，答案可能的最大值就是 x 最大的一半加上 y 最大的一半减去 x 的另一半减去 y 的另一半。大多数可以构造，有些特殊情况稍微变化即可。	O(nlogn)	O(n)
9238	Printer	有 n 个任务，每个任务有个接到的时间 si,需要的工作量是 ti 和优先度 xi。执行的方式每个时刻做当前优先度最高的工作，完成 1 的工作量。现在有个任务的优先度不知道，但是知道它的完成时刻。问这个优先度和每个任务的完成时刻。 n≤50000	很容易发现这个任务的完成时刻和优先度是负相关的。二分答案，然后用堆来模拟即可。	$O(n*\log^2 n)$	O(n)
9232	Wi-fi Towers	平面上有 n 座塔，每座塔有一个范围半径。选择其中若干个，使得这些塔范围内的塔都被选择了。对于每座塔，选择后有一个收益（可负）。求最大收益。 T 组数据 n≤500，1≤T≤55	对于正收益，向汇连边，代价为收益。对于负收益，由源连边，代价为相反数。对于所有覆盖关系，连边，代价为正无穷，对这个图求最小割即可。	$O(n^4T)$	$O(n^2)$
9233	numbers	有 1,2..n 的排列 a1,a2,……,an。您想删除一些整数,使得结果序列满足以下三个条件: 1.由此产生的序列不是空的; 2.序列中所有数异或和等于 0; 3.如果您把所有数按十进制从前往后依次无间隔地写在一行形成一个大的十进制数，这个数将会被 p 整除。 找到一种方案。 1 ≤ n , p ≤ 50000，且 p 是质数。	只使用 1~31 就可以。f[i][j][k]表示用前 i 个数，异或值为 j，模 p 余 k 是否可行。可以判断解，再通过辅助数组来求具体解。	O(32np)	O(32np)
9234	Ping-Pong	每个时刻都有一个区间的集合。您每次可以从集合中的区间(a, b)移动到另一个满足 c < a < d 或者 c < b < d 的区间(c, d)。您需要判断是否有一种从区间 x 到区间 y 的移动方案。 区间长度依次增大。 1 ≤ n ≤ 10^5	将能互相到达的两个线段合并。用并查集维护。 每多一条新的线段找到与其相交的线段，用线段树来做。查询只需要判断是否在一个集合或者被覆盖。	O(nlogn)	O(n)

编号	试题名称	题目大意	算法讨论	时间复杂度	空间复杂度
9235	Polygon	给一个边数 n，构造一个 n 边形，内角相等，边长不重复。 3<=n<=100	n=3,4 –> 无解 其他，以一个点为起点，边不断增加一小点，绕一圈，最后求个交点即可。	O(n)	O(n)
9221	Candies Game	有 n 堆糖果，每一次可以可以选取两堆，假设各有 a，b，a<=b 那么操作成 2a,b-a。问能否最后只剩两堆。 $n<=10^3,\sum a_i<=10^6$	假设有三堆 a,b,c(a <= b <= c)，通过操作可以是其中一堆小于 a。具体做法令 x = b/a。对于 x 的每一位，如果是 1 则将 b 给 a，否从 c 给 a。最后 b 变成 b%a。	O（$n\log^2 n$）	O(n)
9222	Doodle Jump	有 n 个平台。第 x(1≤x≤n)个平台的高度是 a ·x mod p，其中 a 和 p 是互质的正整数。Doodler 最大可能跳的高度是 h。也就是说，它可以从高度 h_1 跳到 $h_2(h_1<h_2)$，如果 $h_2-h_1<h$ 的话。一开始 Doodler 站在高度为 0 的地上。现在的问题是它能不能跳上最高的平台。t 组询问 $1\le t\le 10^4$, $1\le a\le 10^9$, $1\le n<p\le 10^9$, $0\le h\le 10^9$	我们只需要求出最远的两个台子的距离。 如果 a*n<=p，可以直接求解。x = max(a, p-a*n) 否则，假设要 k 次之后第一次取模，然后已这 k 次替换掉原来的 a、n。	O(tlogn)	O(1)
9223	Liars and Serge	有 n 个人坐在一张长桌边上。对于每个人，我们知道他总是说真话或是说谎。 小塞尔吉问他们：你们中有几位总是说真话呢？桌子上的每个人都知道关于桌子上的所有人的所有事情（某个人说真话或是说谎）。诚实的人总是回答正确的答案，而说谎的人会回答 1 到 n 之间除了正确答案以外的任意一个整数。 他拿来一张纸并写下 n 个整数 $a_1,a_2,...,a_n$，其中 a_i 表示第 i 个人的答案。得到这个序列后，塞尔吉断定，桌上至少 k 个人明显说谎了。 求有多少种不同的答案（长度为 n 的答案序列 a）可以得出桌上恰好 k 个人明显说谎了 $n=2^k$，k≤8	如果说的是 k，那么一定是真话，否则一定是假话。一定是恰好有 k 个人说 k。因此，对于其他数 a，只要不是恰好 a 个人说，就不可能是正确的说谎人数。f[i][j][k]表示考虑到 i，j 个回答过，k 个人说实话的方案数。转移很简单。由于 n≤256，队 256 特殊处理(打表)即可。	$O(n^4)$	$O(n^3)$

编号	试题名称	题目大意	算法讨论	时间复杂度	空间复杂度
9224	The Red Button	这个按钮下面的电路由 n 个从 0 到 n-1 编号节点组成。为了关闭这个按钮，这 n 个节点必须以特定的序列拆解。节点 0 必须首先拆解，在拆解了节点 i 后，下一个被拆解的节点必须是(2·i) mod n 或(2·i)+1 mod n。最后一个被拆解的节点必须是节点 0。节点 0 必须被拆解两次，其他节点必须刚好被拆解一次。你的任务是找到一个符合要求的顺序并输出它。 $2<=n<=10^5$	如果 n 是奇数肯定无解。 n 是偶数，正反同时走就 可以构造出解，同时走选的方案不同即可。	O(n)	O(n)
9227	Flights	一个国家，包含 n 个城市。城市的编号从 1 到 n。这里有许多单向的航班让你可以在两个城市之间旅行，但是这些航班的安排使得你一旦离开一个城市就永远不会再回来了。现在要改变一些航班的飞行时间。具体来说，把某些航班的飞行时间改为 2 小时，使得所有从 1 号城市到 n 号城市的旅行路线都花费相同的时间。 $2 \le n \le 1000$，$1 \le m \le 5000$	对于整个图，不断松弛找到一个从原点出发的最短路长度。然后判断是否可行	O(nm)	O(m)
9230	Cleaning Up	有 N 头奶牛，每头那牛都有一个标号 Pi，1 <= Pi <= M <= N <= 40000。 现在 Farmer John 要把这些奶牛分成若干段，定义每段的代价为：若这段里有 k 个不同的数，那代价为 k*k。那总的代价就是所有段的代价总和。求最小值。	如果在某段区间中，不同的数超过了 sqrt(n)个， 那么很显然，我们还不如将整个区间分为 n 段。 那么我们就不用管区间不同数超过 sqrt(n) 的数了。 令数组 b[j]表示 从 b[j] + 1 到 i 不同的数的数量不超过 j 的最左端 然后转移的同时维护 b 数组即可。	O(nsqrt(n))	O(n)
9231	PE lesson	有 n 个人，手上拿着自己编号的球。每一次操作，两个人交换手中的球。已知每个人至多交换 1 或 2 次，问最后持球的方案数。 $1 \le n \le 10^6$	一个合法的持球方式为，将其分解为循环，每个循环至多有 2 个 1 的人。答案只跟 1,2 的个数有关，设为 a，b，将 2 的任意和 1 组合。 Ans(a.b) = (a+b)!÷a!×f(a)然后求 f(a) = f(a-1)+(a-1)*f(a-2)	O(n)	O(n)

编号	试题名称	题目大意	算法讨论	时间复杂度	空间复杂度
9202	Figure Eight	给一个 n*n 的网格图，有些格子有障碍。要你选择一些格子构成’8’: *数字 8 由上下两个矩形构成。 *数字 8 的上下两个矩形都满足至少有一个单元格在矩形内部。 *数字 8 顶部的矩形的底边必须为底部矩形顶边的子集。 *数字 8 只能刻在大理石完美无瑕的部分。 *规定数字 8 的得分为上矩形和下矩形的面积的乘积，它们希望得分能达到最大。 n≤300	枚举 8 的中间，然后只需要求上下分别的最大。用动归求出上面最大的和下面最大的即可。以上面为例，f[i][j][k] 表示考虑到第 i 行，从 j 到 k 为底向上的最长延伸长度。如果(i,j),(i, k)都没有障碍。那么 f[i][j][k]=f[i-1][j][k]+1 否则 f[i][j][k] = -1	$O(n^3)$	$O(n^3)$
9203	Dima and Figure	Dima 喜欢在一块长方形纸片上作自己喜欢的画。每一行选择连续一段涂黑，可以不染。 要求染过色的块是相邻的且任意两个存在直走黑色的路径使得曼哈顿距离恰好为最小。 1<=N,M<=150	有四种情况，左边或者右边是上一行的内部或者外部。 定义 f[i][j][k][w]，表示第 i 行，染从 j 到 k。w 表示两边的前几行的关系。转移压迫分类讨论。	$O(mn^2)$	$O(mn^2)$

编号	试题名称	题目大意	算法讨论	时间复杂度	空间复杂度
9204	race	TOC 是一个长方形的城市，有着 N*M 个方格状的街区。 若干建筑物，双向直线的道路以及交叉路口组成了 TOC。 每个建筑物：占恰好的一个街区。 每条道路：所有道路的宽度都为 1 个街区，并且道路都是水平的或竖直的。 每个交叉路口：占一个街区，位于道路的交汇处。 我们称两个街区是相邻的，当且仅当它们之间存在着公共边。 没有两条道路或两个交叉路口是相邻的。 在每年的狂欢节中，TOP 总会按照一条特殊的路线游行。 TOP 会从一个街区出发，接下来经过若干个交叉路口，最终在一个街区停下。 TOP 知道从某一个街区到其相邻的街区所花费的时间。 同时，从交叉路口需要花费 1 分钟来到达相邻的街区。 TOP 不能经过有建筑物的地方。 我们知道 TOP 的初始与结束位置，也知道 TOP 途中经过的交叉路口的顺序。 但他完成游行之后，他会一直呆在结束位置。 你的问题是，找出在他出发了 K 分钟之后，位于哪个位置。 TOP 总会沿着最短的道路经过给定的交叉路口，而到达终点。 注意到 TOP 可能会访问某些格子多于一次。 3<=m,n<=100,1<=k<=100000	模拟即可，每一次走一步，判断方向。	O(k)	O(nm)
9208	Two Sets	给 n 个数，分成两堆，对两堆求两堆的异或 x1,x2，求 x1+x2 的最大值。给出 x1 最小的方案 $n<=100000,x<=10^{18}$	将所有数异或，考虑某位，如果是 1，那么无论怎么分，x1，x2 这一位一定是一个 1，一个 0。因此优先考虑为 0 的位。用高斯消元判断。	$O(n\log^3 x)$	$O(n+\log^2 x)$

编号	试题名称	题目大意	算法讨论	时间复杂度	空间复杂度
9209	Mirror Room	有一个 n*m 的网格图，有 k 个点是障碍。有一束光从一个格子开始，碰到障碍或者边界会反射。光会进入无限循环，问至少被光束通过一次的空格子数 n，m，k≤100000	首先可以推出，路线第一次循环一定是回到起点。可以证明，反射的次数是O(n+m+k)次，所以只需要快速处理反射就可以。 对于一个斜线上的障碍排序，二分可以找到反射的位置，通过判断旁边格子是否有障碍确定反射的类型。还有一个很重要的性质，任何格子中的光线只有一条。计算答案首先算出这一次循环中每个点被经过了几次。如果没有反向的反射，每个点经过两次，否则 4 次。用总长度除即可。	O((n+m+k)logk)	O(n+m+k)
9211	Triangle Counting	给出 n 个笛卡尔坐标系上的整点，统计有多少三角形包含原点(0,0) 1<=n<=10^5 -10^5<=X_i,Y_i<=10^5	首先转化问题为有多少个不包含。发现原点和三个点的连线的跨度小于 180 度，也就是说两条夹角小于 180 度的射线中间又夹了一条射线，这样的三角形才会是不符合要求的。这样枚举不符合要求的三角形上按极角序出现的第一个点，这时只需按极角序找到两射线夹角小于 180 度的最远的那个点。	O(N)	O(N)
9212	Theft of Blueprints	给出一个 n 个点的带权无向图，满足对于任意一个大小为 k 的顶点集合 S，恰好有一个点与 S 每一个点都有边。令这个点为 v(S)，并且对 S 进行操作的代价是 S 中每个点与 v(S)的边权之和。现在求对于一个大小为 k 的子集操作代价的期望。 1<=k<n<=2000	总方案数为 c(n,k)。然后考虑度数≥k 的点。对答案的贡献为连出去的边的边权和*c(度数-1，k-1)。	O(nk)	O(nk)

编号	试题名称	题目大意	算法讨论	时间复杂度	空间复杂度
9214	Olya and Graph	Olya 现在有一张 n 个点、m 条边的有向无权图。现在我们以某种方式给所有点从 1 到 n 标号，从而保证原图中任意从 u 到 v 的有向边满足不等式：$u<v$。 现在 Olya 想知道有多少种方案添加任意数量（可能是 0）的有向边，使得该图满足下列条件： 1、 从点 i 出发，可以到达点 i+1，i+2,…..,n。 2、任意从 u 到 v 的有向边满足不等式：$u<v$。 3、 两点之间最多有一条边。 4、 对于一对点 i、j($i<j$)，若 j-i<=k，那么从 i 到 j 的最短距离等于 j-i 条边。 5、 对于一对点 i、j($i<j$)，若 $j-i>k$，那么从 i 到 j 的最短距离等于 j-i 或 j-i-k 条边。 我们认为两种添加边的方案不同，当且仅当存在至少一对点 i、j($i<j$)，第一张图中有一条从 i 连向 j 的边，而第二张图中没有。 帮助 Olya。由于要求的答案可能太大，将答案对 1000000007 取模后输出。 2<=n<=10^6,0<=m<=10^5,1<=k<=10^6。	最终的图上只会有 ① a->a+1, ② a->a+k+1 2 类边必须相交。 然后统计方案数。	O(n)	O(n)
9187	Cow Patterns	给定一个长度为 n 的主串和长度为 m 的模式串，如果在主串中存在这样一个子串：子串长度与模式串长度相同，且子串中各个数字的大、小、同关系和模式串中的大、小、同关系是一样的，就称该子串满足条件。求所有满足条件的起始位置。 1<=n<=100,000 ， 1<=m<=25,000 ， 1<=S<=25	对于任意长为 k 的串和 S 的前 k 位对应。新加进来一个数字，只要前面比他小的数字在 S 里对应的也比他小，比他的的数字大在 S 里对应的也比他大，相等的对应相等，就满足要求。修改 KMP 的比较函数即可。对于一个数，找到他前面比他小的最大的，比他大的最小的，和他相等的数的位置，检查。	O(n)	O(n)

编号	试题名称	题目大意	算法讨论	时间复杂度	空间复杂度
9189	Distinct Paths	有一个 n*m 的木板，一些块已经被涂上给出的 k 种颜色中的一种。你需要把每个没涂色的块涂色使得从左上角到右下角的每条路径都不会经过两个颜色一样的块。路径只能向右或向下走。求方案数 n,.m<=100，k<=10	当 k<n+m-1 的时候无解。从上到下，从左到右枚举。每个点不可选的颜色为以这个点为右下角的，（1,1）为左上的矩形的所有颜色	$O(k^{n+m-1})$	O(nm)
9190	Cow Schul	小美正在上学并且分数还不错。她考了 N 次试，第 i 次考试得分为 Ti，而满分为 Pi。 在计算总成绩 G 前，她的老师先将把分数率 Fi 最低的 d 份试卷去掉，其中 Fi=Ti/Pi。 然后计算剩余 Ti 之和以及剩余 Pi 之和，最后计算总成绩 G=(ΣTi)/(ΣPi)。 小美精通数学，所以很快发觉这并没有想象中那么好。小美想告诉她的老师所有满足以下条件的 d：去掉 d 份试卷，她的总成绩 G 可以比老师算出来的更高。 小美很惊讶地发现，没有两次考试分数率是一样的。 1<=N<=50000，0<=Ti<=Pi<40000	首先对于老师的一种方案，应用分数规划的一般做法，求出所有的 c=t-rate*p，如果没有选择的 c 值中的最大值比选择了的 c 值中的最小值大，那么这个解是可以改进的。 那么问题就转化成了怎么求最小的 c 和最大的 c。 t-rate*p 求这种类型的最值，并且 rate 是单调的，那么就可以考虑利用斜率优化的那种办法来维护决策点。	O(nlogn)	O(n)
9192	Transferring Pyramid	一个 n 层的金字塔，第 i 层有 i 个数。覆盖单点需要代价 3，覆盖子塔 i 需要代价 2+size(i)。给定其中 K 个位置，求覆盖这些位置的最小总代价。n、K≤10^5	如果一行一行做，非常复杂。考虑从左到右。 有两个个很重要的性质，各个三角形互不包含，三角形最深不超过 800。 这样 Dp 可以斜着一列一列决策。f[i][j]表示当前是第 i 列，延伸到了 j 的最大权值和。	O(nsqrt(k))	nsqrt(k)

编号	试题名称	题目大意	算法讨论	时间复杂度	空间复杂度
9193	Hill Walk	平面上有 n 个互不相交的线段，看做山。从(0,0)开始的第一座山开始爬，到头后掉落到下一座山，掉落到负无穷就结束，问会经过多少座山。	对所有点排序，从左到右扫描。用一个 set 维护扫描线上的所有山的高低关系。因为山互不相交，高低关系是固定的。每次扫描到一个左端点时，将这座山填进来，右端点时移除，当从山顶掉落，到达的位置就是这座山下面最高的山。	O(nlogn)	O(n)
9194	Polo the Penguin and Lucky Numbers	定义幸运数字是正整数,同时它们的十进制数表示只能包含幸运数字 4 和 7。例如,数字 47, 744, 4 是幸运数字,而 5, 17, 467 不是。 企鹅 Polo 有两个正整数 l 和 r(l < r),他们都是幸运数字。此外,他们的长度(数字的十进制数表示，没有前导零)是相等的。 假设在 l 与 r 之间（包括 l 和 r）有 n 个不同的幸运数字，ai 表示序列里的第 i 个(以升序排列)。请计算出 $a_1 a_2 + a_2 a_3 + ... + a_{n-1} a_n$。答案可能非常大,所以要对 1000000007 ($10^9 + 7$)求余。 $1 \le l < r \le 10^{100000}$	数位递推。 只考虑后 i-1 位的答案与第 i 位的答案的关系 由于只有两个数，在考虑后 i 位时，就只有两种可能，第一位只有 4，或者有一部分 4，后一部分是 7，于是就可以转移了	O(logr)	
9195	Little Elephant and Broken Sorting	给一个 n 个数的排列，和 m 次交换操作。每个交换操作都有 50%的概率交换，问最后的逆序对的数量的数学期望	令 f[i][j]表示 i 比 j 大的概率。初始根据排列，f[i][j] = a[i]>a[j]。每次交换 a,b，f[c][a] = f[c][b] = (f[c][a] + f[c][b])/2， f[a][c] = f[b][c] = 1-f[c][a]， f[a][b] = f[b][a] = 1。 答案就是 f[a][b](a<b)的和	O(n*(n+m))	$O(n^2)$

编号	试题名称	题目大意	算法讨论	时间复杂度	空间复杂度
9196	Cow Run	奶牛们在长度为 M(2 < = M < = 1000000000)的环形轨道上从相同的位置开始跑步。这个游戏会进行 N(1<=N<=14)回合，需要使用 8 N 张卡片,每张卡片上面都写有一个数字 X_i(0 < = X_i < M)。 每回合约翰取出最前面的 8 张卡片，留下前 4 张或者后 4 张，贝茜继续从约翰选出来的 4 张卡片中留下前 2 张或者后 2 张。接着约翰就会让奶牛们跑 R*X_top 的距离，R 表示奶牛们已经跑过的总距离，X_top 表示贝茜留下的 2 张中的第一张，然后贝茜会让奶牛们跑 X_bottom 的距离，X_bottom 表示贝茜留下的 2 张卡片中的第二张。 如果牛最终离他们的起始位置超过 K(0 < = K < =向下取整(M / 2))，他们就无法回到家。 对于每回合,你的任务是确定约翰应该选择哪一半的卡片,来保证不管贝茜如何选择都能够使奶牛回家。	记忆化搜索	$O(2^N)$	O(N)
9199	Lucky Tickets	Gerald 有一个朋友 Pollard。Pollard 对 lucky tickets 十分感兴趣（ticket 是一个数列）。一开始他认为：如果在这些数之间加入运算符号和括号使得最终结果等于 100，那么这个 ticket 就是 lucky 的。但是他很快就分析出了所有的 lucky ticket，所以他决定研究更一般性的问题，即 k-lucky ticket。 对于一个 ticket，如果我们在它一些数的左边或者右边加入一些运算符号(“+”, “-”, “*”)和括号使得最终结果等于 k,那么这个 ticket 就是 k-lucky ticket。 举个栗子，“224201016” 就是 1000-lucky ticket ： $(-2-(2+4))\times(2+0)+1016=1000$ Pollard 参加了一个研究 k-lucky ticket 的组织，他请求你帮他找出 m 个 k-lucky ticket $0\le k\le 10^4,\ 1\le m\le 3\cdot 10^5$	分成两个四位，枚举前四位。然后通过后四位来补充。	O(k)	O(1)

编号	试题名称	题目大意	算法讨论	时间复杂度	空间复杂度
9200	Fetch the Treasure	Rainbow 建造了 h 间密室排成一行,这些密室中有 n 间密室里有宝藏。我们称这样的密室为"宝藏密室"。第 i 间（从 1 开始编号）宝藏密室编号为 a_i，里面的宝藏价值 c_i 美元。 Freda 从第一间密室出发探险。每次她可以向前走 k 间密室或者回到第一间密室。 不过，Rainbow 给了 Freda m 个操作。每个操作是下列三种类型之一： 1. 增加技能 x。即允许 Freda 每次前进 x 间密室 2. 让第 x 间宝藏密室的宝藏价值减少 y 美元。 3. 查询 Freda 能到达的密室中价值最大的宝藏并拿走。如果 Freda 不能到达任何一个宝藏密室那么就认为价值最大的宝藏的价值为 0 并不做任何事。否则拿走这个价值最大的宝藏。如果有多个宝藏密室里的宝藏的价值达到最大，那么取走宝藏密室编号最小的宝藏密室（不必是密室编号最小的）。之后宝藏密室的数量减少一个 h <= 10^18,n <= 10^5, m <= 10^5, k <= 10^4	对于每个模 k 的余数，记下第一个到的点。并维护。用堆来维护宝藏。查询从堆顶开始找。	O(klog(m+n))	O(n+m)
9201	Context Advertising	一个包含 n 个单词的文本。一个正式的广告标语有恰好 r 行，每一行最多包含 c 个字符。潜在的顾客总是喜欢看到很多广告，因此你应该确定文本中能写在标语上的最长的连续单词。标语中一行中的单个单词应该用空格隔开。你可以一次性插入多于一个空格。你不能把单词断开，就是说，每一个单词必须完整的在标语中的某一行中出现。另外，你不能改变单词的顺序，就是说，如果你从上到下，从左到右连续地读标语，你应该得到文本中连续的一段。 1<=n,r,c<=10^6;r*c<=10^6	找到每个单词最前延长多少。然后转化成树状。倍增	O(nlogn)	O(nlogn)

编号	试题名称	题目大意	算法讨论	时间复杂度	空间复杂度
9173	Positions in Permutations	P 是 n 个互不相同且不超过 n 的正整数的一个排列。我们设排列 P 的第 i 个元素为 P[i]，n 为排列的长度。 我们称排列中的第 i 个位置是完美的，当且仅当|P[i]-i|=1。 请求出长度为 n 的而且完美的位置数刚好为 k 的排列数是多少。答案要求取模 10^9 + 7。 $1 \le n \le 1000$, $0 \le k \le n$	dp，记下位置 i，i-1 是否被用过，i 是否被用过。j,完美位置数	O(nk)	O(nk)
9174	Greg and Caves	n*m 的方格图，每一行选择连续一段涂黑，可以不染 先是上一行只能是下一行的子集，然后是下一行是上一行的子集。中间有一行或者若干行是最长的（两个黑点的距离最大），然后往上下两边非递增趋势 求方案数。	f[i][j] 表示前 i 行第 i 行长度恰好是 j 的方案数 g[i][j]表示前 i 行，第 i 行的长度小于等于 j 的方案数	O(nm)	O(nm)
9179	Greedy Elevator	有一幢 m 层的办公楼，其中有一个先进的电梯控制系统，工作方式如下： 所有办公楼层用 1-m 依次编号。在 t=0 的时刻，电梯位于第 1 层。电梯是空的而且别的楼层上没有人在等待。有 n 个人，每个人我们知道三个参数：他开始等电梯的时刻，他初始位于哪个楼层以及他想去哪个楼层。 如果而没人等并且电梯空的，不动，否则去需求多的方向。 模拟电梯的工作并且告诉每一个人他能在哪个时刻到达他想去的楼层。	对上下分别用一个堆来维护。复杂的处理。	O(nlogn)	O(n)
9180	Summer Homework	初始有一个整数序列 a1, a2, …, an。你的任务是对它进行 m 次操作，有以下 3 种： 1. 输入 x 和 v，将 ax 赋值为 v 2. 输入 l 和 r，你需要计算这样一个和： $\sum_{x=0}^{r_i-l_i}(f_x \cdot a_{l_i+x})$ 其中 f0=f1=1，且 i>=2 时， $f_i=f_{i-1}+f_{i-2}$ 3. 输入 l, r 和 d，你需要对 al 到 ar 区间加上数 d	定义一个函数 S(x) $S(x) = F0 + xA_l + F1 + xA_{l+1} + \ldots + Fr-1 + xA_r$。发现 S(x)=S(x-1)+S(x-2)。用线段树，对于每个区间维护 S(0)，S(1)	O(nlogn)	O(n)

编号	试题名称	题目大意	算法讨论	时间复杂度	空间复杂度
9182	Mystic Carvings	一个圆环上有 2*n 个洞穴两两一组，组间有连线。有三对熊，每对熊找一组洞穴。我们定义洞穴 X 和洞穴 Y 的距离为：从洞穴 X 走到洞穴 Y 至少需要经过的其他洞穴数+1（只能沿着边缘行走）问每对的距离相等的方案数。 对于 100%的数据 n<=100000	三条线只有 5 种情况。直接算比较麻烦。求不合法然后减。使用树状数组来求。	O(nlogn)	O(n)
9185	Cows in a Skyscrape	有 n 头奶牛各有重量 wi，电梯限重 l，求最少要多少次电梯让多有牛下来。 n≤18	逐步加大次数 dp[S]，S 表示每只奶牛是否上过电梯，值为是否可行。f[S]表示(dp[i])(i ∈ S)是否都可行。每次维护 f[S]	$O(n*2^n)$	$O(2^n)$
9157	Shaass and Painter Robot	给你一个 N*M 的网格，一开始都是白色的。上面有一个机器人，一开始位于格子（X，Y）上（占据整个格子），面朝某个方向（左上，左下，右上，右下之一）。然后机器人会一直顺着这个方向走下去，每当遇到边界时会遵循光的反射定律改变方向，然后继续走。每当机器人走到一个格子后，它会将这个格子染黑，用掉一个单位颜料。即便这个格子已经被染黑了，也需要耗费一个单位颜料。当机器人意识到这个 N*M 的网格已经变成黑白相间的时候，它会立即停止行动。现在希望你求出，机器人停下来的时候，已经耗费了多少颜料？或者指出永远不可能停下来。 N、M<=100000	模拟。每次走到边界。撞到边界 n+m-1 次就会停止。用 set 维护到过的边界格子。	O((n+m(log (n+m))	O(n+m)

编号	试题名称	题目大意	算法讨论	时间复杂度	空间复杂度
9158	Ace in the Hole	艾米有一副 N 张卡片的牌，牌上面的数值为 1 到 N。 她把这副牌排列好了，且保证没有三张牌的子序列是递减的。本知道，这副牌没有长度为 3 的递减的子序列，但他并不知道确切的顺序。他想找到值为 1 的卡牌。他可以选择任意卡，翻开它，然后重复，直到他发现了卡值 1。在每一步，本选择的卡，将最小化最坏情况下的次数，亦即本的策略并不会浪费他翻开卡的次数。 本之后告诉你，他的运气非常不好，不得不翻开所有 N 张牌才找到卡值为 1 的卡。 给你本翻牌的顺序，求出艾米放置的每张卡的值。 如果有多个可能性，请输出字典最大的。	假设当前还有 p1~pm 没有翻开，剩下的值为 v1~vm。 如果当前翻的是 pk(k<m)，那么这张牌为 v_{k+1} 否则这张牌为 vm，特殊情况为 v_{m-1}：m<=2，或者存在 pm 之前被翻开的数，值在 v_{m-1}~v_m 中间	$O(n^2)$	O(n)
9161	Binary Key	假设有 p 和 q 两个长度为正整数的字符串，我们分别叫它们为匣子和钥匙。 其中，钥匙串 q 只包含字符 0 和 1。 将 q 循环。至长度与 p 相等。取出 q 中为 1 的位对应的 p，即为 S。 给出 p，S 和 q 的长度，求字典序最小的 q。	首先枚举有多少位为 1，接着枚举，判断。	$O(l^2)$	O(L)
9162	Ksusha and Square	Ksusha 碰到一个面积不为 0 的凸多边形。她现在想知道：如果她随机选择两个不同的格点(格点可以在凸多边形内部或者边界上面)，并以它们之间的连线为对角线做一个正方形，那么这个正方形的面积期望是多少？ 其中两个格点 p,q(p!=q)与 q,p 被视作相同。	对 x，y 分开计算。枚举 x，算 y。枚举 y，算 x。	O(n)	O(n)
9165	Photo	有 N 头奶牛排成一排，标号为 1..N。小明拍了 M 张照片，照片 i 包含了从 a_i 到 b_i 的奶牛，每张照片中恰有一头奶牛有斑点。问至多有多少头奶牛有斑点。 n≤2*10^5,m≤10^5	f[i]表示到 i 最多有多少斑点。转移枚举边即可。用单调队列优化。	O(n)	O(n)
9166	Optimize!	有一个数列 a，一个数列 b。对于 i 从 1 到 n，计算 a1~ai 找 b 中配对，使得和大于 h。可以找到答案加一。	首先对 bi 排序。然后 ai 对应一个区间。用线段树维护，每个点记录需要≥bi 的有多少个。	O(nlogn)	O(n)

编号	试题名称	题目大意	算法讨论	时间复杂度	空间复杂度
9167	Best Cow Line	FJ 这年非常的忙碌，需要马上回他的农场，所以他希望能被检查的越快越好。他的奶牛在登记前已经排好了。Fj 决定重新安排他的奶牛。 FJ 选好了新队伍的位置，然后他动手把他的奶牛一个一个从旧队伍中安排到新队伍里面。Fj 不断把剩下的旧队伍中，第一个或者最后一个安排到新队伍的末尾。当他完成的时候，FJ 便会以这个新的顺序去参加登记。 给定初始奶牛的顺序，输出用这个方法可以的到的字典序最小的方案。	贪心。如果两头不等，选小，否则往里比。同时记下之前比过的结果。	O(n)	O(n)
9170	Endless Matrix	有一个矩阵，里面有连续的数字，从 1 开始。 若 $a_{i,j} < a_{t,k}$ $(i, j, t, k \geqslant 1)$ 当且仅当 1. $\max(i, j) < \max(t, k)$ 2. $\max(i, j) = \max(t, k)$ 且 $j < k$ 3. $\max(i, j) = \max(t, k), j = k$ 且 $i > t$ 给定 x_1, y_1, x_2 和 y_2 $(x_1 \leqslant x_2, y_1 \leqslant y_2)$ 求的后 10 位。 $t \leqslant 10^4, x_1,y_1,x_2,y_2 \leqslant 10^9$	直接判断当前区间包含的格子，可以切分为矩形和三角形。分别计算。	O(t)	O(t)
8798	Jeff and Removing Periods	对一个序列操作。一次操作包括，选择一个下标为等差数列且对应元素相等的若干数，删去，重新排这个序列（可以打乱顺序）。 定义一个序列的美好度为删除到空需要的步数。求一个大序列的 q 个连续子序列，每个的美好度。。	可以看出来，第一次操作后，序列中不同数的个数是之后的操作次数。所以只需要求序列中不同数的个数和是否可以在第一次消除所有的某数。将询问按 r 分类。只要知道某个数最后出现的位置，通过 1~r 的数确定。然后用树状数组求出(l,r)里包含了多少种。对于等差数列计算每个数位截止的等差数列的开头是谁，记为 f[i]。然后知道某个数最后出现的位置 i，如果（l,r）能够包含某个(f[i], i)，第一次就可以消除所有的这种数。同样用树状数组求。	O((q+n)log n)	O(n+q)

编号	试题名称	题目大意	算法讨论	时间复杂度	空间复杂度
8800	Tournament-graph	给 n（n≥3）个点的无向完全图的边定向，要求任意两点距离不超过 2。	4 个点的时候不行。 假设 k 个点的时候可以。对于 k+2 个点的时候，在 k 个点的基础上加上 1~k -> k+1, k+1 -> k+2, k+2 -> 1~k，那么新图符合条件。 n=3 和 n=6 是点为奇数和偶数的最小满足条件的 n。手解出这两种情况即可。	$O(n^2)$	$O(n^2)$
8801	Land Acquisition	Farmer John 要买 N（1≤N≤50000）块长方形的土地，每块土地长为 Li，宽为 Wi（Li,Wi 均为正整数且≤1000000）。 购买单独一块土地的价格是$1/平方，但是购买多块土地有优惠。如果一次性购买几块土地，总价钱就是这几块土地的 max{Li}*max{Wi}。 Farmer John 想通过分次购买，花最少的钱把所有土地都买下来，问他最少要花多少钱。	按 L 递减排序。然后 W 是递增的。(去掉被完全包含的)然后 f[i]表示买前 i 块的最小价钱。用斜率优化。	O(n)	O(n)
8802	Playing with String	两个人轮流行动，不能操作的人输。 游戏开始前裁判买老师会在方格纸上写下一个字符串，每个格子包含一个字母。 一个人的操作分这么几步： 1.这个人选择一张纸，我们称上面写着的字符串为 t。注意一开始时只有一张可选纸。 2.这个人选择一个 i(1<=i<=|t|)使得存在一个正整数 k(0<i-k,i+k<=|t|)满足 t[i-1]=t[i+1] and t[i-2]=t[i+2] ... t[i-k]=t[i+k] 3.这个人以把这张纸的第 i 个字母的两侧撕开使得这张纸分成 3 份:t[1~i-1],t[i~i],t[i+1,|t|]	首先发现所有可选择点是固定的，不论如何操作，不会新增可选择点。可选择点两边相等即可。 只看可选择点。分为若干段。互不影响。求长度为 i 的连续可选择点的 SG 函数，异或起来即可。	$O(n^2)$	O(n)

编号	试题名称	题目大意	算法讨论	时间复杂度	空间复杂度
8803	Have You Ever Heard About the Word?	一个重复块(repeating block)由一个字符串与自身连接而成，如 abcabc 是一个重复块，而 abcabd, ababab 不是。 你有一个由拉丁字符组成的字符串。每一步你要找到它的子串中最短的重复块，如果有多于一个，你必须选择最左边的那个。你要将那个形如 XX(X - 某个字符串)的重复块替换成 X，换句话说你要删除其中的一个 X。重复以上步骤直到字符串中不存在重复块。 求最终的字符串 1<=输入字符串长度<=50000	随着删除，重复块的长度递增。要找长度为 n 的重复块的。没 n 个点标记，重复块一定跨过其中的一个点。求过这个点的重复串长度。用哈希快速求。	O(nsqrt(n))	O(n)
8804	Google Code Jam	Vasya 打算去做 GCJ，他可以在比赛开始后快速（忽略所有读题时间）读题。然后对于 n 道题目，他知道以下 5 个值。 1、scoreSmall_i 表示的是做出简单部分之后得到的分数。scoreLarge_i 指的是做出困难部分后得到的另外分数。 2、 timeSmall_i 表示的是做第 i 题的简单部分所花的时间。timeLarge_i 表示从简单解法，优化到困难的解法的所需额外时间。 3、Vasya 是一个身经百战的 coder。所以他一旦写完简单部分的解，就可以 1A。然而对于困难部分，他有 probFail_i 的概率 FST(fail system test)。并且值得注意的是，如果 FST，那么就不会得到困难部分的分数，并且这次提交也不算一个正确解的提交。 比赛总共有 t 时间。读题和提交题的时间忽略不计。并且值得注意的是：Vasya 不允许在一道题目的其中一个部分中提交两次解。 Vasya 想要正确安排时间和解题顺序，然后使得得到的期望分数尽可能高，同时保证期望罚时尽可能少。 $(1 \leqslant n \leqslant 1000, 1 \leqslant t \leqslant 1560)$	首先考虑，决定好做哪些题的哪些部分，通过调整得到的期望值是不一样的。排序。f[i][j]表示前 i 道题，j 的时间的期望。枚举下一个题目的决策来转移。	O(nt)	O(nt)
8805	Sereja and Squares	给的若干左右括号，求括号匹配数方案。	f[i][j]表示到了 i，左括号比右括号多 j 个要用滚动数组。	$O(n^2)$	O(n)

编号	试题名称	题目大意	算法讨论	时间复杂度	空间复杂度
8806	Balance	给定一个由 n 个注水容器组成的系统。某几对容器由输水管道相连接。通过管道，你可以将整数升的水在其连接的容器间进行传输（管道都是双向的）。两容器间可能连着多于一根管道。管道的数量为 e。每个容器的容积为 v 升。显然，在传输过程中任一容器中的水量不能超过 v 升。 给定每个容器初始状态的水量 a[i]和目标状态的水量 b[i]，请求出一个输水方案实现这一目标。输水的总步骤数不能超过 2*n^2。 1<=n<=300,1<=v<=10^9,0<=e<=50000	首先随便求一棵生成树，然后在树上做。枚举每个点，将这个点多出的全部移动到少了的里面	$O(n^2)$	O(e)
8807	Tape Programming	有一种编程语言，在这个语言下一个程序是由数字和”<”,”>”构成的非空串。程序运行时有一个指针。最开始指针的指向最左字符，移动方向为向右。 我们重复以下操作直到指针指向串外： 1、如果指针指的位置是一个数字，输出这个数字，然后将指针沿着原来移动方向移动，同时将原来的数字减一。如果原来的数字为 0 则删除这个数字，串的长度减一； 2、如果指针指的位置是”<”或”>”，那么指针的移动方向对应得改为向左或向右（与符号的尖角方向相同），接着指针沿着新的移动方向移动。如果新的位置也是”<”或”>”，则删除原来的”<”或”>”字符。 现在我们有一个由 n 个由"<", ">"和数字构成的串 s1, s2, ..., sn ，你需要回答 q 个询问。每个询问会给你两个数 l,r，如果把 sl, sl + 1, ..., sr 看出一个单独的程序，问你每个数字会被输出多少次。	一个串，运行过的区间永远是从起点开始的连续一段。因此挑出来的任何一个子串[l,r]的运行过程，就是运行主串的时候第一次进入 l，到从[l,r]出去的过程。因为第一次进入 l 时的串和子串相同。子串运行结束对应着运行的指针从[l,r]出去。 对于一个子串[l,r]的求解，转化为在运行主串的序列中找到第一次 l 的出现位置，以及这之后第一次从[l,r]中出去的位置。通过前缀和作差可以求。。	O(n)	O(n)
8808	Rotatable Number	他想要找到最大的 b(1 < b < x)，满足在 b 进制下存在一个长度为 n 的正“可旋转数”（允许有前导零）	一个可旋转数可以表示为 $\frac{b^{p-1}-1}{p}$,b 是进制，p 是个质数不是 b 的约束	O(n)	O(n)
8809	Number Challenge	定义 d(n)为 n 的约数个数。现在，你有三个数 a,b,c。你的任务是计算下面式子 modulo 1073741824 (2^30)的值。 a,b,c<=2000	逐个枚举每个质数，然后将问题规模缩小。	$O(a^2)$	$O(a^2)$

编号	试题名称	题目大意	算法讨论	时间复杂度	空间复杂度
8810	Yaroslav and Algorithm	Yaroslav 喜欢算法。我们将描述一个他最喜欢的算法。 1.这个算法接受一个字符串作为输入。我们设这个输入字符串为 a。 2.这个算法由一些命令组成。i 号命令的形式为"s[i]>>w[i]"或"s[i]<>w[i]"，其中 s[i]和 w[i]是长度不超过 7 的字符串（可以为空），由数字或字符"?"组成。 3.这个算法每次寻找一个编号最小的命令 i，使得 s[i]是 a 的子串。如果没有找到这样的命令，那么整个算法终止。 4.设找到的命令编号为 k。在字符串 a 中，s[k]第一次出现的位置会被 w[k]替换。如果这个命令形如"s[k]>>w[k]"，那么这个算法继续执行（译注：回到第 3 步）。否则，算法终止。 5.算法的输出就是算法终止时字符串 a 的值。 Yaroslav 有一个 n 个正整数的集合，他需要一个这样的算法，且能够使每一个数加 1。更正式地，如果我们把每个数看成一个十进制表示的字符串，那么对于每个字符串独立地运行这个算法，这个算法需要输出一个输入串对应的数+1 的字符串。	做法比较神奇。有通用解法。 思路首先在开头插入个?,做指针用。然后移动至尾部。然后让位数加 1（不是 9）。由于分阶段，所以开始用两个？，然后到尾部替换为 1 个？。来标记当前在什么。9 结尾比较麻烦，一步一步做进位。	O(1)	O(1)
8783	Close Vertices	你得到了一棵包含 n 个点的树，树上的每条边有一个非负边权，树上两点间路径的长度是该路径包含的边数，树上两点间路径的权重是指该路径包含的边的边权之和。 我们说两点是“相邻”的，当且仅当，存在一条连接该两点的路径，满足该路径的长度小于等于 L,且权重小于等于 W。 统计有多少个点对(u,v)，满足 $u < v$，且 u,v 是相邻的。	树分治。选择重心后，按 l 排序，然后用树状数组求出符合 w 的个数。	$O(n\log^2 n)$	O(n)

编号	试题名称	题目大意	算法讨论	时间复杂度	空间复杂度
8784	White, Black and White Again	有 n 天，有 w 件好事和 b 件坏事。每天至少发生一件事，并且只会都发生好事或坏事。事情的发生顺序是连续若干好事，之后连续坏事，最后又是若干好事。求可能的方案数。 3<=n<=4000，2<=w<=4000 1<=b<=4000，w+b>=n	首先乘 w!,b!，转化为无序。然后将坏事找个空进去，然后在分成 n 天	O(n+w+b)	O(1)
8786	Year of More Code Jam	新的一年又开始了。 在这一年里，有几场对 little Josh 来说意义重大的线上赛。虽然这些线上赛的开始日期还没有确定，但是主办方都已将日程安排公布（即线上赛开始后的每一天是否有比赛）。 在某些情况下，一天可能有多场比赛。，对于每一个拥有 S 个比赛的日子，little Josh 会获得 S^2 点愉悦值。little Josh 的初始愉悦值为 0。 这一年有 N 天，每场线上赛的开始日期等概率地分布在每一天上（即：在每一天开始的概率都是 1/N）。现在最大的问题就是 little Josh 在这一年的期望愉悦值是多少。 用带分数 K+A/B 来表示他的期望愉悦值 假如一场线上赛的某次比赛的日期在第二年，那么不应该被算入总愉悦值内。 1<=N<=10^9,2<=m<=50,1<d[2]<d[3]<...<d[m]<=10000,T<=50	很容易发现分母为 n^2 然后发现 d[i]<=10000，先计算前 1w 天然后再算最终结果。平方需要拆开来算。运算溢出比较麻烦。	O(10000m)	$O(m^2)$
8787	Ciel and Flipboard	Ciel 有一个 n 行 n 列的板子，每个格子上有一个数字。 大家都知道 n 是一个奇数，不妨设 x=(n+1)/2。Ciel 可以选择一个 x 行 x 列的子矩阵，并将其中的所有元素乘-1。他可以使用这个操作任意多次。 Ciel 的目标是最大化板子上的数字和。 n<=33	枚举第一行，然后从左往右从上往下贪心。	$O(n^2 2^n)$	$O(n^2)$

编号	试题名称	题目大意	算法讨论	时间复杂度	空间复杂度
8791	Maxim and Calculator	一个计算器有两个整数单元，一开始，第一个单元包含数字 1，第二个单元包含数字 0。 这个计算器支持一以下两种操作： 1.假设第一个单元的数字为 a，第二个单元的数字为 b，那么将第二个单元的数字改成 b+1。 2.假设第一个单元的数字为 a，第二个单元的数字为 b，那么将第一个单元的数字改成 a*b。 现在 Maxim 想知道，有多少个正整数 x(l<=x<=r)满足，存在一种方式从计算器初始状态开始，操作不超过 p 步之后使得第一个单元中的数字为 x 2<=l<=r<=10^9,1<=p<=100.	可能到达的状态不会太多。预处理出来，然后在这些状态间转移。	O(500000+rp)	O(500000)
8793	Colorful Stones	现有两串彩色的石头。每块石头的颜色是红、绿、蓝之一。给你两个颜色字符串 s 和 t。 初始时 Squirrel Liss 站在第一串的第一个石头上，Cat Vasya 站在第二串的第一个石头上。你可以发出下面的指令（0 次或多次）。 每个指令是以下三种中的一种：“红色”、“绿色”、“蓝色”。在发出指令 c 后，站在颜色为 c 的石头上的动物将会移动到后面的一块石头上。 (Liss 的位置,Vasya 的位置)被称为状态。如果从初始状态(1,1)能够通过发出 0 次或多次指令达到一个状态，我们就称这个状态为“可达”的。请计算出不同的可达状态的数目。 N <=1000000	考虑第一个小动物站在位置 i，第二个小动物可能的位置形成了一段区间。 特殊情况不可达： ….xy，……yx	O(n)	O(n)
8794	Candy Store	有 n 个人，每个人会买不超过 C 的糖果。你需要预先买好盒子，大小数量任意。保证可以对于每个人都满足要求。求最小数量。1 <= T <= 100，1 <= n <= 1000，1 <= C <= 10^12。	贪心，每一次找一个当前需要的最小盒子，直到可以满足所有要求。	O(Tlog(C*n))	O(1)

编号	试题名称	题目大意	算法讨论	时间复杂度	空间复杂度
8795	Xenia and Dominoes	一个拼图是一个 3 × n 的桌子，除掉一些禁止块(forbidden cells)，并包含多米诺骨牌。覆盖满，只有一个空位。 要完成一个拼图，你需要用若干步把空格子从开始位置移动某个指定位置。一步移动是在保证拼图合法的情况下，把一个多米诺骨牌移到空格子里。横向的多米诺骨牌只能横向移动，纵向的多米诺骨牌只能纵向移动。你不能旋转多米诺骨牌。 Xenia 有一个 3 × n 的带禁止块和一个圆圈标记(circle-marked)的格子的桌子。Xenia 还有很多完全一样的多米诺骨牌。现在 Xenia 想知道，如果她把多米诺骨牌放在桌子上，能有多少种不同的合法的拼图。同时，Xenia 要求圆圈标记的格子没有被覆盖。这个拼图还必须至少能移动一次。 3<=n<=10^4	枚举是否能上下左右动，然后放好某些块，左状态压缩 dp，然后用容斥原理。	O(n)	O(n)
8796	Dima and Game	Dima 在纸上写下 n 对整数(l[i],r[i])(1<=l[i]<r[i]<=p)。然后玩家轮流进行操作。轮到自己时，可以进行下面的操作： 1. 选择第 i 对数(1<=i<=n)，满足 r[i]-l[i]>2; 2. 将第 i 对数替换为(l[i]+floor((r[i]-l[i])/3),l[i]+2*floor((r[i]-l[i])/3))或者(l[i] , r[i]-floor((r[i]-l[i])/3))。floor(x)表示向下取整。 不能进行操作的玩家则输。 Dima 希望先进行操作的 Anya 赢得游戏。所以 Dima 需要写下这样的 n 对整数(l[i],r[i])(1<=l[i]<r[i]<=p)，使得如果两个玩家都采取最优策略，先操作的玩家取得胜利。请计算 Dima 有多少种这样的方法 1<=n<=1000,1<=p<=10^9	求出 SG 函数，并计算出一个 SG 函数对应的方案数。f[i][j]表示放了 i 对，SG 函数为 j 的方案数。	O(n)	O(n)
8797	Counting Skyscrapers	有若干大楼，每座摩天大楼的高度（即楼层数）是被独立地随机选择的：对于每个正整数 i，楼层数为 i 的概率为 2^{-i}。 为了加快中转运输的效率，摩天大楼间修建了一些滑索。一座摩天大楼的第 i 层和另一座摩天大楼的第 i 层之间有滑索当且仅当两楼之间没有摩天大楼有第 i 层。 A 计算楼的准确数。B 通过绳索，忽略绳索下的楼，计数器加 2^h。给其中一个人的计数器，求另一个人计数器的期望值。	如果给的是 B 的，A 和他一样。如果是 A，可以通过数学计算得到简单的式子。	O(nh)	O(h)

编号	试题名称	题目大意	算法讨论	时间复杂度	空间复杂度
8768	Three Swaps	有一个 1 到 n 顺序的数列。做 3 次区间反转，给你后来的序列求这三次操作反转的区间。1<n<=1000	实际上只有至多 7 个点有意义。然后 dfs	$O(n^2 2^7)$	O(n)
8769	Tennis Rackets	一种正三角形的球拍，设计完外形后你要在这个框架上穿线。正三角形的每条边上有 n 个小孔，把该边分成了等距的 n+1 段。每条边上，离每个顶点最近的 m 个小孔是用来改善气流的通风孔，因此线不能穿过这些孔。源于一种创新的理念，球拍网需要连成一个钝角三角形，三个角分居正三角形框架三条不同的边上。请你帮忙统计，有多少种可行的三角形球拍网设计方案。 1≤n≤32000，0≤m≤n/2	枚举第一条边，再枚举第二条边，第三条边对应连续一段区间。随着第二条边的变化而变化。	$O(n^2)$	O(1)
8770	Donkey and Stars	在平面直角坐标系上的第二象限有 n 个点。初始在原点。有两条射线，角度为 a1,a2。每一次从当前点发出的射线夹角内找到一个点，问做多能动多少次。 1≤n≤10^5， 0°≤α1<α2≤90°	首先转化坐标，然后问题转化为最长上升子序列。	O(nlogn)	O(n)
8772	Meeting Her	n 个点，m 条边的有向无权图。有若干公交车，走 li 到 ri 的其中一条最短路径。要从 a 到 b，可以中途上下任何一辆公交车。求最少需要乘坐公交车的次数。 1 <= n <= 100	首先 floyd 求出最短路，然后判断每辆车必经的点。然后动归算出最少次数。	$O(n^3)$	$O(n^2)$
8775	The Evil Temple and the Moving Rocks	n*n 的网格图，有若干石头，分为上下左右。选择一块石头，它会按自己的方向移动。碰到另一块会停止，激活碰到的石头。石头移动并且碰撞就会发出声音，要求出声次数超过 p。求构造。n≤300，p≤n^3-n^2	构造方法如下： 奇数行，前一半是右，最后一个是下，后一半隔一个有一个右。 偶数行对应。	$O(n^2)$	O(n)
8778	Maximum Waterfall	给出一些墙，水从高往低流，每次只能到达一面墙，选择一个路径，使得路径上的流量的最小值最大。 1<=n<=10^5,2<=t<=10^9	按高度排序之后，将所有的坐标离散化。 按高度将线段插入到线段树中，线段树中维护这个区间最高的一条线段的标号，也就是标号的最大值。 用 dp 做一次最优解	O(nlogn)	O(你、)

编号	试题名称	题目大意	算法讨论	时间复杂度	空间复杂度
8779	Tower of Hay	N 摞干草（标号 1 到 N）在传送带上有序的进入牛栏。第 i 摞干草有一个为整数的宽度 w_i；所有干草摞有一个单位的高度。 Bessie 必须用全部的 N 摞干草去建造那个塔而且必须按照它们到达的顺序放置它们。她可以按照她的愿望放置随意多摞干草在一行上。接下来她可以放置接下来到达的干草摞在上次排成的一行上面来建造新的一行（但不能比下面的行要宽）。直到所有的干草被用完。她必须按干草到达顺序来堆叠它们。更清楚地说：一旦她将一个草包放在第二级，她不能将接下来的草包放在地基上。 1<=N<=100000, 1<=w_i<=10000	动归，从后往前做，f[i]表示到 i 位置塔低的宽度，g[i]表示此时的高度。用斜率优化。	O(n)	O(n)
8782	Roadside Trees	在一条直线有 n 个位置可以种树，自西向东标号 1~n。每个月，每棵树都会长高 1 米。每个月初，你都会收到一个要求。要求有两种类型： 1、 在位置 p 种一棵高度为 h 的树 2、 砍掉从西向东数的第 x 棵树，当这个位置的树被砍掉后，倒下的树会占据这个位置，之后这个位置不能再种树。 在做完这个要求后，你需要求出当前树高的最长上升子序列。高度不会重复。 1 ≤ n ≤ 105；1 ≤ m ≤ 2 ·105；1 ≤ pi ≤ n； 1 ≤ hi ≤ 10；1 ≤ xi ≤ 10	树的高度相对不变，可以忽略生长，这样每棵树变成一个平面上的点。 求最长上升子序列。由于树是会删除和添加，但是不会删除和添加太多。暴力删除添加即可。	O(mlogn * (max(h,x)))	O(n)

编号	试题名称	题目大意	算法讨论	时间复杂度	空间复杂度
8753	Paradox Sort	你需要给这些蜡烛排一个序，按这个顺序扔给 foreseeable 一根蜡烛，每次 foreseeable 他获得一根蜡烛(除了第一根)，他就会与他手上的蜡烛相比较，如果两者之间他更喜欢新得到的，他就会扔掉手上的蜡烛，用新得到的蜡烛替换，否则他就扔掉新蜡烛，手上的蜡烛不变。 当所有蜡烛都扔完后，foreseeable 就会收藏他最后手上的那一根。 foreseeable 不一定有最喜欢的蜡烛，我们知道对于任意一对蜡烛 a,b，他更喜欢哪一个。你想让 foreseeable 最后收藏一根特定的蜡烛。给定 foreseeable 对于蜡烛两两之间的喜好关系。确定是否存在一个蜡烛的排列让 foreseeable 最后点起你想要他点的蜡烛，如果有解，请输出字典序最小的排列。 1<=N,T<=100 0<=A<100	因为要求字典序最小，所以逐位枚举，判断。 把喜好关系看成有向图，然后做 dfs，删去当前点能直接到达的点和 dfs 树上的点，如果还有点则无解，否则有解。	$O(Tn^3)$	O(n)
8755	The Great Julya Calendar	给一个数 n，每次可以减去它包含的一个数字，问变成 0 最少减多少次。 $n <= 10^{18}$	将这个数的首位取出，然后把剩下的减掉，首位变小。不断执行即可。	$O(\log^2 n)$	O(1)
8756	Wall Bars	1、我们定义单位长度，建筑物中间那根管子高度为 n。 2、在高度为 1、2...n 的地方，恰好有一根水平的横杆从中间的杆子连向四个方向中的某一个预先固定好的杆子上。 3、如果两根横杆的距离不超过 h，且方向相同，那么一个孩子可以从一个一根横杆爬到另一根上。在地上的孩子，可以爬到任何一根高度在 1-h 之间的横杠上。在 Manao 的建筑物上，一个从地面出发的孩子至少能到达一根高度在 n-h+1、n-h+2...n 的横杠。 Manao 想知道有多少种设计方案满足上述要求。 1≤n≤1000, 1≤h≤min(n,30)	f[i][j][k][w][q]表示前 i 行，三个杆高度为 j,k,w，q 表示另一个杆是否符合要求，此时的方案数。转移枚举放在哪一边。	$O(nh^3)$	$O(nh^3)$

编号	试题名称	题目大意	算法讨论	时间复杂度	空间复杂度
8757	Tourists	*在一些时刻会有 2 个游客同时从点（-1,0）和（1,0）出发（去散步）。其中第一个人出发点（-1,0）,第二个人出发点为（1,0）。 *每一对游客都以相同的速度 1 行动（每秒前进单位长度）。第一个人沿直线 x=-1、第二个人沿直线 x=1 行动。他们都向 y 轴正方向移动。 *在一些时刻墙会出现。墙（li,ri）是一条在点（0,li）和（0,ri）之间的线段。每个墙都瞬间出现。 Ultima Thule 官方想要了解对于每一对同时出发的游客,他们将会有多长时间无法彼此望见？ 帮助他们计算所要求的时间。	坐标化之后，问题就变成求一堆的梯形之间的覆盖关系， 开一个堆，记录当前最靠左的梯形，之后的所有梯形的 t 都当作这个最左的梯形，如果遇到梯形的 r 点，则从堆删去他。 然后预处理前缀合，二分就行了。	O(nlogn)	O(nlogn)
8761	Monsters and Diamonds	Piegirl 发现一只怪物和一本关于怪物和馅饼的书。当她在读这本书的时候，她发现有 n 种怪物,每种都有一个唯一的 1 到 n 的编号。如果你喂怪物一块饼，它就会分成一定数量的怪物（可能为零），以及至少一个多彩的钻石。怪物们可能存在多种分裂方式。 最开始 Piegirl 有且只有一只怪物。她先喂了一块饼，它随之分裂。对于分出来的怪物，继续喂饼，直到它们都分裂为钻石。然后她把所有钻石收集起来。 对于每种怪物，确定以其为起始，Piegirl 可以得到的钻石最少最多分别是多少。Piegirl 有无限多的饼。 1≤m, n≤10^5	首先拓扑排序。找到环，判断这个环是否能够无限产生钻石。剩下的部分按拓扑序做即可。	O(n)	O(n)
8762	First!	给若干字符串，问可以任意改变字典顺序的情况下，哪些串可以变成最小的。 1≤n≤30000，1≤字符串总长度≤300000	建立 trie 树，然后对于每个串构造一个 26 个点的有向无环图，拓扑排序判断是否有环。	O(L)	O(L)
8763	Two permutations	你有两个各包含 n 个元素的排列 p 和 q，和 m 个由 l1，r1，l2，r2 组成的询问。每次询问在 p 中位置在[l1,r1],在 q 中位置在[l2,r2]中的数的数量。 一个 n 元素的排列是指 n 个不同的数，每个数在 1-n 之间。 数字 v 的位置在排列 g1,g2,……，gn 中是使得 gi=v 的 i。	首先按第一个序列的下标重标第二个序列，问题转化为 q 中[l2,r2]中包括多少个[l1,r1]。用可持久化线段树。	$O(n\log^2 n)$	O(nlogn)

编号	试题名称	题目大意	算法讨论	时间复杂度	空间复杂度
8765	Rats	n*m 的网格图，有若干格子有老鼠和障碍，你可以放两个炸弹，炸弹可以炸掉距离为 d 内的老鼠。问是否能够炸完所有老鼠。1<=n, m<=1000，1<=d<=8	随便选一个老鼠，枚举可以炸到它的位置，在找另一只还活着的，再枚举。	$O(d^8)$	O(nm)
8766	Cubes	给出 n*n 个格子，每个格子上有若干个单位立方体。从无穷远有一片光线 V，问最多能照到多少个立方体	枚举四个顶点，找到点积最小的点，按照这个与 V 的点积排序。对于 V 的极角排序。然后用线段树	O(nlogn)	O(nlogn)
8767	Cow Tennis Tournament	给出 n 个人，每个人有个战力值，战力大的获胜。现在有 k 个区间，如果两个人的战力都在区间内，则会更改交战结果。问最后有多少个三元组(i,j,k)满足,i>j,j>k,k>i。	交补转化，C(n,3)是总共的三元组数目，求出有多少个不满足的。 不满足的肯定是其中 1 个人战胜了 2 个人。那么做法就是枚举这个人，判断他能战胜 m 个人，那么就应该减去 C(m,2) 用线段树来维护。	O(nlogn)	O(nlogn)
8738	Rhombus	有一个大小为 n×m 的表，定义一个点 x,y 的函数 f(x,y,k)，其中 x>=k,x+k<=n,对 y 同理，对于所有点 ij 距离这个点曼哈顿距离为 t(t<k)，贡献 aij*(k-t)，给定 k，求最大的 f(x,y,k)和此时所有的 x,y 1<=n,m<=1000 $1 \leq k \leq \lfloor \frac{min(n,m)+1}{2} \rfloor$	对于每个点，计算出这个值，找到最大的就可以。用前缀和预处理斜线方向。 更好的做法，将原图形切成四个三角形，每个三角形单独求。通过平移，差出的部分为一个斜线结束一列系数和距离有关。斜线可以直接求，和距离有关的部分变成一维。通过前缀和求。	O(nmk) (可以过) $O(nm+k^2)$	O(nm) O(nm)
8739	Yaroslav and Arrangements	定义一个序列是良好的，当且仅当相邻的数差的绝对值为 1，首尾是相连的，并且第一个数是所有数里最小的。定义一个序列是优秀的，首先数字单调不减，并且通过重新排列可以构造出至少一种，至多 k 种良好序列。 求长度不超过 n，数字不超过 m 的优秀序列有多少。 n，m，k≤100	f[i][j][k][w]表示考虑到数字 i，此时序列长度为 j，有多少个是 i-1，此时方案数。每次将 i 和 i-1 添加到 i-1 的旁边。	$O(n^3m)$	$O(n^3m)$

编号	试题名称	题目大意	算法讨论	时间复杂度	空间复杂度
8740	Reclamation	有一个 r*c 的地图，把左边界和右边界粘起来使得形成一个圆柱，现在要不断地挖去其中的格子，要求任何时候都存在一条从最上方到最下方的路径(四联通)，如果某次操作不满足要求则不做，问最后有多少次操作是成功的。r,c<=3000，n<=300000	将地图放大为 2r * c 维护挖去的格子之间的八连通性。挖 x,y 的同时等价于挖掉 r+x,y，如果相连则断开否子可以操作。	O(n)	O(n)
8741	The Last Hole!	Luyi 在平面上放了 n 个圆，第 i 个圆的圆心在(xi,yi)。最开始所有圆半径都为 0，然后所有圆同时开始变大，在时刻 t(t>0)所有圆的半径都为 t。我们可以想象成一些黑色的实心圆放在一个无穷大的白色平面上，每个时刻都会存在一些黑色和白色的联通快。需要注意一点，随着圆的增大，越来越多的圆会相交。 我们定义一个白色的封闭区域为一个’洞 ’，求最早的时刻，使得之后再也没有洞。0<n<=100, -10^4≤xi,yi≤10^4	枚举其中圆心构成的三角形或者矩形，找到一个最大时刻。	$O(n^4)$	O(n)
8742	Cow Neighborhoods	每只奶牛都有一个不同的坐标。两只奶牛是属于同一个群的，当且仅当至少满足下列两个条件之一： 1. 两只奶牛的曼哈顿距离不超过 C，即 | x1 - x2 | + | y1 - y2 | <= C。 2. 两只奶牛有共同的邻居。即存在一只奶牛 k，使 k 分别与这两只奶牛同属一个群。 给出奶牛们的位置，请计算有多少个牛群，以及最大的牛群里有多少只奶牛。1 <= N <= 10^5	将每个点变为（x+y,x-y），用一个队列，维护队列中元素 x 坐标的差小于 c，不满足则左指针右移对这个队列中元素的 y 坐标维护 set，如果新加入元素的前驱后继与它的 y 坐标差值不超过 c，则用并查集将他们连在一起	O(nlogn)	O(n)
8743	Dividing Kingdom	平面上有 n 个点，你需要横竖各切两刀，分成就块，并且九块内包含的点恰好为 a1~a9，顺序任意。(9 <= n <= 10^5)	枚举顺序，然后可以直接判断切的位置，再判断合法。	O(9!logn)	O(9!logn)
8744	Piglet's Birthday	Winnie 的家里有 n 个架子，每个架子上都有一些蜜罐。初始时所有蜜罐都是装满蜜的。Winnie 一共去了 q 次架子；第 i 次 Winnie 会先去第 ui 个架子，拿走 ki 个蜜罐，把这些蜜罐中的蜜吃掉，然后把这些蜜罐都放到第 vi 个架子上。Winnie 想知道，每次操作后，架子上所有蜜罐都被吃完的架子的期望个数是多少(1 <= n <= 10^5，0 <= ai <= 100)。	f[i][j]表示第 i 个架子上有 j 个的概率。每一次被操作过的两个架子的蜜罐总数是确定的。因此转移枚举 j。	O(a*n)	O(a*n)

编号	试题名称	题目大意	算法讨论	时间复杂度	空间复杂度
8745	Colorado Potato Beetle	一个机器人要移动 n 次，每次选择一个方向移动若干步。到过的地方会喷杀虫剂。问最后有多少格子是安全的，被杀虫剂围住算安全。$1\leqslant n\leqslant 1000$, $1\leqslant x\leqslant 1000000$	离散化坐标，然后模拟。最后做 flood-fill	$O(n^2\log n)$	$O(n^2)$
8746	BerDonalds	给定一个无向带权联通图，求图的直径, n<=200，c<=100000	只可能在边的端点或者中点。 floyd 求最短路。 枚举每条边，然后计算	$O(n^3)$	$O(n^2)$
8750	More Queries to Array...	你有一个包含 n 个整数 a[1],a[2],...,a[n] 数组。你的任务是快速地执行以下两种操作。 1.将闭区间[l,r]中的元素赋值为 x。即此操作过后 a[l],a[l+1],...,a[r]的值都为 x。 2.计算$\sum_{i=l}^{r} a_i \cdot (i-l+1)^k$并输出和，此处 k 不超过 5。因为这个和可能很大，你要输出它模 1000000007(10^9+7)的结果	$(i-l+1)^k=[i-(l-1)]^k$ 用线段树，记下 0~5 次方。然后求的时候用二项式定理。	O(nlogn)	O(n)
8751	GCD Table	定义一个 n*m 的表格，aij = gcd(i,j)。给你一个长度为 k 的数列，问是否在表格中为一行连续出现	首先可以确定行号一定是整个数列的最小公倍数 M。然后设第一个数的列为 x，得到一个模线性方程组。整理这个方程组后用中国剩余定理解。然后判断 x 是否满足要求。在[1,M]里 x 是唯一的，可以通过 x 判断是否有解。	O(klogM+sqrt(M))	O(k)
8752	k-Maximum Subsequence Sum	给一个序列，支持点修改，和求最大 k 个不相交子段和。1<=n,m<=10^5, k 不超过 20	维护最大子段和。每次找到最大，然后将这一段反转继续做。	O(knlogn)	O(n)