A. Battle of Balls

BFS

球的半径为 r，为了不让球碰到障碍点，就要保证，球心到障碍点距离大于 r。

问题可转化为，每个障碍点有一个半径为r的圆，需要知道从地图底边到地图顶端是否联通。

对于任意两个障碍点或一个障碍点和一条左右边界，如果它们之间的距离小于球的直径，则球通不过它们之间的间隔。

把所有距离小于 $2 \times r$ 的障碍点连一条无向边；右边界和左边界两条直线抽象成两个点 s, t，把与其距离小于$2 \times r$ 的障碍点连一条无向边，得到一张无向图。

如果地图不连通，则一定存在一条从地图左边界 s 到右边界 t 的路径。

通过 BFS 就能知道s 与 t之间是否存在一条路径，即可知道地图底边和顶端是否联通

时间复杂度 O(n^2)

B. Icebound and Sequence

$S = (\sum_{i=1}^{n} q^i) \, mod \, p \; (1 \le n, p, q \le 10^9)$

简单想法直接模拟，复杂度 $O(n)$ ，n 太大会 TLE

solution 1

考虑等比数列的性质，如果我们想要计算$S = q^l + q^{l+1} + \cdots + q^r$，我们可以先计算$S' = q^0 + q^1 + \cdots + q^{r-l}$，然后整体乘上$q^l$,即$S = S' * q^l$。

利用这个性质，我们使用递归分治法解题。假设递归的这一层我们需要计算$T = (\sum_{i=1}^{b} q^i) \, mod \, p$，我们先递归计算$T' = (\sum_{i=1}^{\lfloor b/2 \rfloor} q^i) \, mod \, p$，将得到的结果乘上$1 + q^b$，则得到$S = T' * (1 + q^b) = q + q^2 + \cdots + q^{\lfloor b/2 \rfloor * 2}$。因为最后一项为向下取整再乘二，如果$b$为奇数的时候还需要再加一个$q^b$。全程注意取模!

对于每一层我们都像上面那样递归，每一层都需要使用快速幂计算，共有$logn$层，总复杂度为$O(logn * logn)$，如果我们每层都记录一个参数$Q = q^b$，,则无需快速幂，总复杂度为$O(logn)$

solution 2

令$s[i]$表示等比数列前i项的和，则$s[i] = s[i-1] * q + q$，这是一个线性常系数递推式，可以使用矩阵快速幂处理递推。

solution 3

根据等比数列求和公式：当$p \neq 1$时，$S = (\dfrac{q^{n+1} - 1}{q - 1} + 1) \, mod \, p$

由于 p 可能为合数，$(q-1)$在模 p 意义下可能不存在逆元

所以要改变一下模数，先将模数改为 $p' = p \times (q-1)$ ，计算模 p' 意义下的答案

$S = a \times p' + b$，由于 S，a 能被 $(q-1)$ 整除， 故 b 能被 $(q-1)$ 整除，就可以不用求逆元

C. 分治

区间 dp

用 $dp[i][j]$ 表示攻打完第 i 个国家到第 j 个国家共 $(j-i+1)$ 个国家需要的最小花费 第一层循环枚举区间长度，第二层循环枚举区间左断点，第三层循环枚举最先攻打区间 $[i,j]$ 内的城市k。则状态转移为 $dp[i][j] = min\{dp[i][k-1] + dp[k+1][j] + (j-i) * cost[k]\}$ 复杂度是 $O(n^3)$

D. 榜单

模拟 + I/O

给出一场比赛的所有提交记录，输出比赛最后的榜单

正确理解题意，字符串格式化输出

需要注意CE不计入榜单和罚时；一道题目AC后的提交不在统计

E. Paper Plane Fly Away

给一张二分图，求每条边与其他边有多少交点

对于一条边可用区间 $[l,r]$ 表示。对于求交点数可转化为，所有区间按照左端点排序后，右端点的逆序数，加上按照右端点排序后，左端点的逆序数

使用两次经典的分治或者树状数组即可求解

F. Take Apples

使用搜索打表找规律，或结合博弈论相关知识可以确定：

当$M = k * (s+1), N < s+1$时，Bob存在必胜策略。

否则Alice存在必胜策略。具体证明详见另一个文件。

G. 点我

签到题，直接模拟或找规律。

注意 $n = 0$ 时的情况

H. 天神的密码

本题数据范围最大可以出到$N=10^{1000000},K=10^{18}$。接下来按数据范围分别讨论做法:

当$N \le 10^9, K \le 2$时:

直接模拟即可。注意N^K不要使用pow函数计算，pow返回的是double类型，会掉精度。

当$N \le 10^9, K \le 500$时:

因为$(10^9)^{500}$只有4500位，使用高精度乘法模拟即可。

当$N \le 10^{1000000}, K \le 10^{18}$时:

考虑这个过程的性质，设某个数N的第i个数位的数字为$N[i]$,可以知道:

$N \, mod \, 9 = N[1]*10^k \, mod 9 + N[2]*10^{K-1} \, mod 9 + ... + N[k] \, mod 9 = N[1]+N[2]+N[3]+...+N[k]$

也就是说，对于某个数字N来说，他的所有的数位上的和对9取模等于数字N对9取模。

设第i次操作后的值为y_i,我们可以知道$N^K \, mod \, 9 = y_1 \, mod \, 9 = y_2 \, mod \, 9 = y3 \, mod \, 9 = ... = y_n mod \, 9$

由此，可以得知：题目中对于数字$X=N^K$进行变换的过程等价于令N^K对9取模,使用快速幂计算即可。

I. Twinkle

对于 t 时刻的询问，答案为 $\sum((s_i+t) \; mod \; (c+1))$

t 时刻时，初始值 $s_i \ge (c+1-t)$ 时，值需要减去 $(c+1)$

问题转化为 $sum\_s + num \times t - cnt \times (c+1)$

划分为三个问题

1. 二维平面，求一个矩形中多少个点（num）
2. 二维平面，求一个矩形中所有点的权值之和（sum_s）
3. 二维平面，求一个矩形中有多少个点的权值 $< x$ （cnt）

前两个问题等价，都是二维偏序问题，排序后使用树状数组或 CDQ 分治，$O(nlogn)$

第三个问题是一个三位偏序问题，一维排序，二维CDQ分治，三维树状数组 $O(nlog^2n)$

总复杂度 $O(nlog^2n)$

J. 舔狗

给出有向基环树森林，最多选取多少个没有公共顶点的边?

solution 1

找到环后，拆环后进行两边树形 dp

$dp[x][0]$ 表示以x为根的子树，不选择它的边的答案数

$dp[x][1]$ 表示以x为根的子树，选择它的一条边的答案数

$dp[x][0] = \sum_{y \in son[x]} max\{dp[y][0], dp[y][1]\}$

$dp[x][1] = \sum_{y \in son[x]} max\{dp[y][0], dp[y][1]\} + max_{y \in son[x]}\{-max\{dp[y][0], dp[y][1]\} + dp[y][0] + 1\}$

solution 2

贪心

从所有入度为 0 的点BFS，贪心选择它指向的点，并把它指向点指向的点的度数减一，当点的入读为 0 时，加入队列；如果指向的点被选过答案加。

对于剩下的节点只可能存在环上，统计环上点的个数，两两配对。个数为奇数答案加一。

K. 河北美食

模拟+字符串

用Map处理菜品名称，直接模拟即可。注意按要求输出

L. Smart Robot

假设答案的长度为k，即机器人需要走$k-1$步。则机器人在矩阵中最多产生$N \times N \times 4^{k-1}$个数字。

则答案一定小于等于$N \times N \times 4^{k-1}$，即$10^k \le N \times N \times 4^{k-1}$，化简可得：

$k \le \log((N \times N)/10) + 1$ 其中对数是以 2.5 为底的。

因为N最大为 50，所以答案的长度一定很小，最高不超过7 ,则机器人最多走6步。

所以直接搜索，限制每个点最多搜索6步，使用哈希表记录，最后筛选出答案即可。

但是事实上答案并不会达到上界，而且题目是随机生成的，所以只需要走4步即可。