Problem A. 理财

定位：easy

签到题。直接输出 $n/p*q-n$ 即可。

Problem B. 棋盘

定位：easy

n, m都是偶数的时候是无解的，考虑棋盘黑白染色，考虑一条路径一定是从黑点走到白点再走到黑点，n, m 都是偶数的时候黑点和白点个数差 2，因此无解。考虑如何构造解，当n为奇数的时候，从 $(1, 2)$ 出发，一直走到 $(1, n)$ 然后走到 $(2, n)$，再反向走，走到 $(2, 1)$ 以此类推，按照 S 型走，最后一定可以走到 $(n, m-1)$，如果 n 是偶数 m 是奇数，将横纵坐标倒过来，再按照之前构造的方法即可。

时间复杂度 $O(n \times m)$，可以通过本题。

Problem C. Cover Master

定位：medium-hard

首先特判掉 $k = 1$ 的情况。

考虑什么时候能让覆盖的区间个数变少，当一个线段树上的区间的左右区间都有覆盖的时候，我们可以把它左右延长，让它覆盖整个区间，这样可以减少覆盖的区间个数。

我们可以分别处理出向左和向右延长，使得覆盖区间减少个数为 x，所需要延长的长度，记为 a_x, b_x，之后我们枚举 a 序列，用总的需要减少的个数减去 a 序列，即可求出对应的 b 序列的位置。对答案取 min 即可。

每组数据的复杂度为 $O(\log n)$，总复杂度为 $O(T \log n)$。

Problem D. Dessert Theif

定位：medium-hard

首先先将每行每列最大值保留，其他的格子糖果数拿到 1。

令 $MaxR[i]$ 表示第 i 行的最大值，$MaxC[i]$ 表示第 j 列的最大值。则若 $a_{i,j} \neq 0$，且 $MaxR[i] = MaxC[j]$，则可以将第 $a_{i,j}$ 变为 $MaxR[i]$，这样就可以获得 $MaxR[i] - 1$ 的贡献.

因此，若 $MaxR[i] = MaxC[j]$ 且 $a_{i,j} \neq 0$，则可以将第 i 行与第 j 列连边，跑一个二分图最大匹配即可求出最优答案。

时间复杂度 $O(n^3)$，可以通过本题。

Problem E. 求导

定位：easy

ax^b的导数为abx^{b-1}，注意题目中多项式的正确表达方式即可。

时间复杂度为 $O(|S|)$，可以通过本题。

Problem F. 蒸汽朋克

赛前出题人和验题人的定位：medium

实际过题情况：hard

对于本题，大部分选手都能够想到直接 DFS 判断，若存在一个连通块，其中没有点被选中，则答案为 oo。问题在于一种特殊情况，对于一个连通块，若它为一个非二分图，则有且仅有一种合法方案——所有齿轮角速度均为 0。

复杂度 $O(n+m)$，可以通过本题。

Problem G. Grand Ark Bible

定位：medium-hard

一道比较经典的期望 DP。令 P_i 表示连续进行 i 次寻访且恰好在第 i 次获得干员的概率，则有：

$$P_i = \begin{cases} p(1-p)^{i-1} & i \le m \\ \\ (1-p)^m(p+(i-m)q)\prod_{j=1}^{i-m-1}(1-p-jq) & i > m \end{cases}$$

同时令 $F[i]$ 表示一共进行了 i 次寻访，且第 i 次刚好获得干员，获得干员数的期望数，$G[i]$ 表示一共进行 i 次寻访，且第 i 次刚好获得干员的概率。则有：

$$F[n] = \sum_{i=0}^{n-1}(F[i]+G[i])P[n-i]G[n] = \sum_{i=0}^{n-1}G[i]P[n-i]$$

复杂度 $O(n^2)$，可以通过本题。注意要讨论 $q=0$ 以及若干次后 $p+(i-m)q>1$ 的情况。

Problem H. 信号传输

定位：easy-medium

首先二分答案，设二分到的值为 x，之后 dp，设 $dp[i]$ 表示到第 i 个城市的优质指数的最大值，则$dp[i] = a_i + \max_{j=0}^{i-x}(f[j])$，设 $g[i]$ 为 $0 \sim i$ 中 $dp[i]$ 的最大值，则 $dp[i] = g[j] + a[i]$，最后与 W 比较即可。

总时间复杂度$O(n \log n)$，可以通过本题。

Problem I. In Hogwarts

定位：hard

要求 S 串所有的后缀与询问串 T 的最长公共前缀之和。我们不妨先考虑如何快速求 S 的某个特定后缀与询问串的最长公共前缀。

注意到后缀自动机中 parent 树里两个节点 lca 表示两个节点代表字符串的最长公共后缀，那么我们可以考虑将 S 串翻转后建立后缀自动机，在建立时就标记该后缀所代表的节点，如果我们能找到 T 在后缀自动机中所表示的节点即可快速求答案。其中存在一个问题，就是如果无法找到 T 所对应的节点，也就是说当发生了失配时，其实我们需要的只是 S 任意一个子串与 T 最长的匹配前缀即可，因此可以直接跳到 parent 树的祖先后继续匹配，只需要在匹配的过程中记录当前匹配的长度即可。

如果要求所有后缀的贡献，其实就是在 parent 树中标记 n 个节点然后一起求 lca 的贡献。从 T 对应的节点开始暴力跳父亲节点即可。

势能分析一下该算法的复杂度，T 对应节点表示的最长字符串长度不会超过 $|T|$，每次跳父亲节点后长度至少减少 1，计算某个节点的复杂度是 $O(1)$ 的，故询问上的复杂度为 $O(\sum |T_i|)$。预处理出后缀自动机是 $O(n)$ 的，总复杂度为 $O(n + \sum |T_i|)$，可以通过本题。

该题也有使用SA的线性做法，此处不再赘述。带 log 的算法也可以通过本题，但需要选手在实现时注意常数。

Problem J. 罚时算术

定位：easy

判定一下首字符是否为负号，负号直接输出0。否则字符串处理一下读入的两个数，计算即可。

如果是使用手写快读来直接读入的选手，可能需要注意特判一下 ‘-0‘ 的数据，样例中也已经给出。

时间复杂度是 $O(n)$ 的，可以通过本题。

Problem K. Kate and Company Management

定位：medium-hard

首先把每个数排序，然后对每个数进行质因数分解，我们依次枚举每个质因子，把具有这个质因子的数依次连边。因为 a_i 的质因子个数小于 $\log a_i$，因此总边数是 $O(n \log C)$ 级别的，其中 C 为 a_i 的值域。

之后我们可以 dp，设 $dp[i]$ 表示到第 i 个的时候最长序列的长度，则 $dp[i] = 1 + max_j(dp[j])$，其中 i 和 j 有连边，答案就是 $max_{i=1}^{n}(dp[i])$，时间复杂度为$O(n \log C)$，可以通过本题。

Problem L. 变进制四舍五入

定位：medium-hard

这是一个构造题。我们只考虑比较特殊的操作 $f(k, 2)$ ，它的意义其实就是将当前的数 N 变成离 N 最近且为 k 的倍数的数，特殊情况就是有两个倍数一样近时，取较小的那个。

对于数 N，考虑如何最大限度地将其变大和变小。

如果执行操作 $f(2N-1, 2)$，那么就可以将其变成 $2N-1$，这样我们可以利用大约 $\log_2(y-x)$ 个操作达到目标。

如果执行操作 $f(\lceil \frac{2N}{3} \rceil, 2)$，那么就可以将其变成 $\lceil \frac{2N}{3} \rceil$，这样我们可以利用大约 $\log_{1.5}(y-x)$ 个操作达到目标。

因此，当值域范围 $C \leq 10^9$ 时，最多仅需要 51 个操作，可以通过本题。