

Educational Codeforces Round 58 解题报告

SGColin

目录

1	A. Minimum Integer	2
1.1	Description	2
1.2	Solution	2
2	B. Accordion	2
2.1	Description	2
2.2	Solution	2
3	C. Division and Union	2
3.1	Description	2
3.2	Solution	2
4	D. GCD Counting	3
4.1	Description	3
4.2	Solution	3
5	E. Polycarp's New Job	3
5.1	Description	3
5.2	Solution	3
6	F. Trucks and Cities	4
6.1	Description	4
6.2	Solution	4
7	G. (Zero XOR Subset)-less	4
7.1	Description	4
7.2	Solution	4

1 A. Minimum Integer

1.1 Description

CodeForces 1101 A

q 次询问，每次给出 l, r, x ，求 x 的最小不在 $[l, r]$ 内的倍数。

1.2 Solution

暴力判断一下 x 在不在区间里，若不在答案就是 x ，否则是 $(\lfloor \frac{r}{x} \rfloor + 1) \times x$ 。

2 B. Accordion

2.1 Description

CodeForces 1101 B

定义手风琴是一个字符串，其两侧分别为“[:”,”:|”中间再加上若干个（可以为 0）”|”，例如“[:|||:]”，给出一个字符串，删掉若干字符，使得留下来的是一个手风琴，且它的长度最长。

2.2 Solution

找到左侧第一个“[”和右侧第一个“|”，然后再在这个区间里找到左右侧第一个“:”，若这四个位置都存在且互不相同，则有合法解。此时把两个“:”之间的“|”全选上即可。

3 C. Division and Union

3.1 Description

CodeForces 1101 C

给出若干个区间，询问是否能把这些区间划分成两个区间集合，使得两集合内的区间并无交。

3.2 Solution

按左端点排序，扫描一遍，记录当前已扫描过的区间最大右端点。如果当前位置的左端点大于维护的最大右端点则有合法解，可以以当前位置为分界线。

左端点相同的区间右端点扫描的顺序是无关的，因为这些区间之间必定有交（左端点）。

4 D. GCD Counting

4.1 Description

CodeForces 1101 D

给出一棵树，点有点权。求最长链长度，满足链上所有点权求 $\gcd > 1$ 。

4.2 Solution

虚树做法：显然 \gcd 不为 1，需要是一个质数的倍数。我们对所有点权分解质因数，然后考虑每个质数的答案，对包含该质数作为因子的点建虚树，然后每个连通块求个直径。

树形动规 1：发现并不用建虚树，打个标记，每次 dfs 子树之前先判断一下当前子节点点权是否是当前质数的倍数，若是就直接跑树的直径，复杂度 $O(n \log n)$ 。

树形动规 2：记录每个节点的质因数分解出的所有质因子，进行重标号，直接跑树的直径，枚举当前点的质因数和子节点的质因数，若相同就转移改质因数对应的数组，复杂度 $O(n \log^2 n)$

5 E. Polycarp's New Job

5.1 Description

CodeForces 1101 E

有两种共 q 个操作：

$+ a b$ 把一个 $a \times b$ 的卡片放入集合。

$? r c$ 问当前集合的所有卡片是否都能放到一个 $r \times c$ 的包里。视为包的厚度是无限的，卡片可以旋转，但放入包中必须边界线平行于包。

5.2 Solution

结论题。对于每张卡片，将它的 $\min(a, b)$ 那一维放到 $\min(r, c)$ 那一维中，较大的边放到 $\max(r, c)$ 的哪个方向上，一定是最优策略，该策略下无解则无解。

尝试证明这个结论。如果一张卡片长的边放到包短的一侧，短的边放到包长的一侧比原来优秀，那么我们完全可以将它旋转 90° ，此时 $\min(r, c)$ 的一维放入的边变短了一定合法， $\max(r, c)$ 的一维放入的边一定也能放下，因为 $\max(r, c) > \min(r, c)$ ，而 $\min(r, c)$ 都能放下。

6 F. Trucks and Cities

6.1 Description

CodeForces 1101 F

有一条公路，所有城市直线排布在公路上，每个城市到路尽头的距离分别为 $a_i m$ 。

有 m 辆货车，每辆从 l_i 号城市行驶到 r_i 号城市，耗油速度是 c_i/m ，开始油箱是满的，之后在每个城市都可以加油，每次都加满，但至多加 k_i 次。

要求所有车的油箱大小统一，问要求所有车加油次数合法所需要的最少油箱大小。

6.2 Solution

二分答案：验证即为暴力。每来一辆车就重新二分一下，但下界是上一辆车的答案，显然答案随着车的假如是单调不降的，复杂度 $O(nm \log k)$ ，使用如上二分方法速度非常快。

区间动规：标程的做法。先不管车，考虑将数列分成 k 段的问题。设 $f[l][r][k]$ 表示将区间 $[l, r]$ 分成 k 段，最长的一段最小值。那么转移就是枚举最后一段的开始位置 p ：

$$f[l][r][k] = \min(f[l][r][k], \max(f[l][p][k-1], dis[r] - dis[p]))$$

这样做复杂度是 $O(n^4)$ 的。但是考虑固定 l, k 那么 $f[l][l+1][k] \dots f[l][n][k]$ 的转移其实是有决策单调性的，因为最后一段变长肯定没什么好处。于是记录转移点复杂度就变为 $O(n^3)$ 。

然后扫描所有车就可以了，答案为 $\max\{f[l_i][r_i][k_i+1] \times c_i\}$ 。

7 G. (Zero XOR Subset)-less

7.1 Description

CodeForces 1101 G

给出一个数列，要求把数列划分成若干段，每个段的权值就是段内所有数的异或和。

如果所有段的某一个子集中，所有段的权异或和为 0 则不合法。求最多能划分成多少段。

7.2 Solution

维护数列的前缀异或和 sum_i 。因为 a_n 必定被包含在某一段里，所以先把 sum_n 插入到线性基里。然后做法是从后往前依次尝试把每一个前缀插入线性基，答案是最后线性基里基底的个数。

这样划分的道理是，每次成功插入了一个前缀，就相当于划分出了从当前位置到上一次成功插入的位置这一段。这样问题转化为一定要选 sum_n ，然后选尽量多的数使得没有一个子集异或为 0。这个东西显然就是基底的个数。