第十八届同济大学程序设计竞赛 简要题解

A 自适应树游走协议

签到题，按照题意模拟即可。使用堆式存储，则叶子结点的编号就是n到$2n-1$。

B.简单的数学题

先套路地用一下莫比乌斯反演

$$\begin{aligned}&\sum_{i=1}^{n}\sum_{j=1}^{i}\lceil\frac{i}{j}\rceil[gcd(i,j)=1]\\&=\sum_{i=1}^{n}\sum_{j=1}^{i}\lceil\frac{i}{j}\rceil\sum_{d|i,\ d|j}\mu(d)\\&=\sum_{d=1}^{n}\mu(d)\sum_{i=1}^{n/d}\sum_{j=1}^{i}\lceil\frac{i}{j}\rceil\end{aligned}$$

然后数论分块，μ的前缀和可以杜教筛，所以只要快速求

$$f(n)=\sum_{i=1}^{n}\sum_{j=1}^{i}\lceil\frac{i}{j}\rceil$$

分两类

1. $\lceil\frac{i}{j}\rceil <= \sqrt{n}$时：枚举$\lceil\frac{i}{j}\rceil$的值$w$，$\lceil\frac{i}{j}\rceil\le w$的点对$(i,j)$数就是$\sum_{i=1}^{n}\lceil i/w\rceil$, 这个值是$w$个1，$w$个2，$w$个3……,可以$\mathrm{O}(1)$求。
2. $\lceil\frac{i}{j}\rceil > \sqrt{n}$时：此时$j<\sqrt{n}$,可以枚举$j$，对于一个固定的$j$，$\sum_{i=1}^{n}\lceil\frac{i}{j}\rceil$是$j$个1，$j$个2，$j$个3……，可以$\mathrm{O}(1)$求，同时去掉$\lceil\frac{i}{j}\rceil$小于$\sqrt{n}$的。

为了保证复杂度是$\mathrm{O}(n^{\frac{2}{3}})$的，可以预处理$n\le 10^6$时$f$的值（类似杜教筛）

C 困难的数学题

对于n的任意一种划分$x_0+x_1+\cdots+x_t$，考虑第一个数x_0：若其大于k，则将其减一之后得到的$(x_0-1)+x_1+\cdots+x_t$恰好是$n-1$的一种划分。若$x_0=k$，直接去掉x_0，剩下的数恰好是$n-k$的一种划分。因此可以得到划分方案数的递推式：$f_n=f_{n-1}+f_{n-k}$，线性递推即可。

D 平衡的字符串

首先，k为奇数则显然无解。对于k为偶数的情况，考虑子串$S_{1..k}$与$S_{2..k+1}$，若它们都平衡，即有$\sum_{i=1}^{k}[S_i=0]=\sum_{i=1}^{k}[S_i=1]=\sum_{i=2}^{k+1}[S_i=0]=\sum_{i=2}^{k+1}[S_i=1]=\frac{k}{2}$。因此可知$S_1=S_{k+1}$。同理，可以得到对于任意$i\le n-k$，都有$S_i=S_{i+k}$。

于是我们可以将S中的字符按照下标对k的余数分为k组，每组中只要存在一个已确定的字符，则所有的'?'均可以被确定。若存在两个已确定的不同字符，则可以断定无解。将所有能确定的字符确定后，考虑前k个字符，只要其中0和1的个数均未超过$\frac{k}{2}$，那么显然可以将其平衡。

E 不平衡的字符串

我们首先注意到下界$\frac{a}{b} \geq \frac{1}{2}$，而我们要求字符个数与串长之比严格大于下界，因此，任何满足约束的子串一定是存在一个字符的数量超过串长一半的。所以，不会有任何一个子串同时满足多个约束，我们完全可以对每个约束分别计数、最后相加。由于串中只有小写字母，因此不同的约束不会超过26个。

对每个约束，记p_i为$S_{1..i}$中s的个数，则串$S_{i..j}$满足约束的条件为：

$$\frac{a}{b} < \frac{p_j - p_{i-1}}{j-(i-1)} \leq \frac{c}{d}$$

我们将约束拆成两个不等式分开考虑，移项可得：

$$\begin{aligned} aj - bp_j &< a(i-1) - bp_{i-1} \\ cj - dp_j &\geq c(i-1) - dp_{i-1} \end{aligned}$$

因此，对每个约束，我们针对$a_i - bp_i$与$c_i - dp_i$分别使用权值树状数组维护，即可在$\mathrm{O}(\log n)$的时间内查询到以i结尾的符合条件的子串个数。注意，由于a, b, c, d的值可能较大，我们需要对其进行离散化。总时间复杂度为$\mathrm{O}(nm\log n)$。

F 值钱的项链

显然，对于每一个位置的候选珠子，只有红色和蓝色的最大价值的珠子有作用，令$\mathrm{val}(i, 0/1)$表示第i个位置蓝/红色珠子的最大价值，不存在则为负无穷

假如项链不是环状的，那么这是一个经典的线性动态规划：令$\mathrm{dp}(i, 0/1)$表示前i个珠子，第i个是蓝/红色的最大价值，转移方程为：

$$\begin{aligned} \mathrm{dp}(i,0) &= \max(\mathrm{dp}(i,0), \mathrm{dp}(i,1)) + \mathrm{val}(i,0) \\ \mathrm{dp}(i,1) &= \mathrm{dp}(i,0) + \mathrm{val}(i,1) \end{aligned}$$

现在项链是环状的，可以破环成链，会影响到的只有第1个和第n个均为红色珠子的情况，于是可以做两次，分别讨论第1个位置是否为红色珠子的情况：

- 第1个位置是红色珠子，答案为$\mathrm{dp}(n, 0)$
- 第1个位置是蓝色珠子，答案为$\max(\mathrm{dp}(n,0), \mathrm{dp}(n,1))$

最终答案为两者中的较大值，时间复杂度$\mathrm{O}(nm)$。

另外值得注意的是，题目要求为"不存在两个相邻的红色珠子"，因此当$n = 1$时实际上是没有约束的，需要加以特判。

G 自行车调度

这题可以使用费用流求解。

首先，若自行车总数不能整除点数，显然无解。

设每个点的自行车的平均数量为w， 对于$a_i > w$的，我们建立从源点到i的边,容量为$a_i - w$,费用为0，对于$a_i < w$的，我们建立从i到汇点的边,容量为$w - a_i$,费用0， 原图的边建拆成两个有向边，容量无穷大，费用为边权。

则该图上的最小费用流即为所求。

由于没有保证图连通，所以要判一下是否满流，不满流则无解。

H 三阳开泰

数位dp，显然条件有6个：三个数的前几位是否取到上界，以及a^b，a^c，b^c的前几位是否达到X的上界

令$\mathrm{dp}(i, j, k)$表示，取前i个二进制位，三个数的上界取到情况为j，异或的上界取到情况为k，那么j和k的取值范围都是$0 - 7$

转移上，枚举三个数在第i位的取值，然后分别考虑自身是否达到上界，以及异或后是否达到上界，直接转移即可

时间复杂度$\mathrm{O}(60 \times 2^3 \times 2^3 \times 2^3 \times T)$

I 图中修边

首先给出本题的结论：

我们称一个图为**坏图**，当且仅当它满足以下两个条件之一：

1. 该子图是一个完全图而且图的大小大于等于3
2. 该子图每一个点的度数均为偶数且图的大小大于等于4

两个条件均不满足，则其为**好图。**

如果一个图的任一连通分量为坏图，则不存在解法。否则，必然存在解法。

证明：首先，由于一次操作不会改变任何一个点的度数的奇偶性，所以如果条件2在操作之前成立，那么操作之后也会成立，反之亦然。而当条件1成立时，无法进行任何操作。

因此，坏图无法通过操作转变成好图，但是一个好图总存在一个操作使得操作之后是一个或者多个好图。

同时，我们可以注意到下述结论： **如果好图进行一个操作之后产生了一个坏图，那么同时一定同时存在一个好图被产生了**

这个分类讨论一下可以得到，在好图里面产生坏图的操作一定会改变原图的连通性。同时，如果存在这样的操作，那么同时一定存在一个操作使得整个图的连通性不变。（因为坏图一定是联通的，仔细画个图想想)

所以，在有限次操作之后，点的度数会不断减少，坏图最终会定留在某个完全图无法分解，好图最后会符合条件。

另外，本题可以在$\mathrm{O}(m)$时间内构造出解，读者可以自行思考。

J 下围棋

本题需要SG函数的前置知识。

对于每棵树，考虑其根节点的所有子树。这个游戏可以看做以子树为石头堆的Nim游戏。因此，其SG函数的值为所有儿子的SG函数异或和再加一（有额外的一个直接在根结点上放棋子的后继）。

整个树的根节点不能放棋子，因此初状态的SG值是根的所有儿子的SG函数值异或和。