Virgil's NOIP 提高组模拟赛

Day2 题解

本套题目主要考察选手数学证明功底和对数学知识的应用能力，思维难度较大。

1. Teleport (teleport.cpp)

> *给定一棵树，每个人可以覆盖与之相邻的节点，求最少放多少人，使得所有点被覆盖。*
>
> $n \le 10^6$

算法壹

输出 $\lfloor \frac{n}{2} \rfloor$ ，时间复杂度 $O(1)$, 可获得 30 分。

算法贰

经典动态规划

$f[i][0]$ 表示以 i 为根的子树中所有点均被覆盖且 i 上无人所需的最小人数（ i 被其儿子覆盖）

$f[i][1]$ 表示以 i 为根的子树中所有点均被覆盖且草地 i 上有人所需的最小人数

$f[i][2]$ 表示以 i 为根的子树中除 i 点以外其余点均被覆盖所需的最小人数

时间复杂度 $O(n)$, 可以获得满分。

算法叁

其实有很多动态规划的题目可以用贪心来解决（笔者就曾用玄学贪心算法通过一道树形 DP 难题），本题也不例外。

贪心遍历每个点，如果它的孩子都没有放人，同时他自己和他父亲都没有放人，就在它的父亲上放人，统计答案即可。

时间复杂度 $O(n)$, 可以获得满分。

2. Flowers on Head (flower.cpp)

> *给定一个带权无限图，每次随机选择一条边走，问从 1 号点走到 n 号点的期望时间。*
>
> $n \le 500, w_i \le 1000$

算法壹

模拟，暴力计算期望，可以获得 30 分。

算法贰

设点 i 到终点的期望为 f_i , 点 i 的度数为 d_i ,

那么 $f_u = \sum_v \frac{f_v + w_{u,v}}{d_u}, f_n = 0$

然后递推 f_1 就可以了。

时间复杂度 $O(n)$ ，可以获得另外的 20 分。

算法叁

成了一个环之后就不能递推了，可以手玩方程组。

时间复杂度 $O(n)$ ，可以获得另外的 20 分。

算法肆

列出 n 元一次方程组，高斯消元求解。

要注意精度问题，考虑化为 $d_u f_u - \sum_v f_v = \sum_v w_{u,v}$ ，这样方程组就是整系数的了，精度问题得以解决。

时间复杂度 $O(n^3)$, 可以获得满分。

3. Terminal Password (password.cpp)

> *给定正整数数列* a *, 定义* $f(x)$ *为数列* a *中为* x *约数的数的个数*
>
> *定义递推数列* $p_i = p_{i-1} + p_{i-2} + f(i-1)$ *，求* p_m *.*
>
> $n \le 10^5, m \le 10^9, a_i \in [2, 10^9]$

算法壹

递推+模拟，时间复杂度 $O(n^2)$, 可通过 $1 \sim 6$ 个测试点，获得 30 分。

算法贰

矩阵快速幂求斐波那契数列。

设矩阵 $F_i = \begin{bmatrix} f_i & f_{i-1} \\ f_{i-1} & f_{i-2} \end{bmatrix}$, 发现 $\begin{bmatrix} f_i & f_{i-1} \\ f_{i-1} & f_{i-2} \end{bmatrix} \cdot \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} = \begin{bmatrix} f_{i+1} & f_i \\ f_i & f_{i-1} \end{bmatrix}$

所以 $F_i = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}^{i-1}$

时间复杂度 $O(log_2n)$, 可以通过 $7\sim12$ 测试点，加上算法壹，获得 60 分。

算法叁

我们可以先考虑把斐波那契数列先用矩阵快速幂求出 fib。

因为 $p_i=p_{i-1}+p_{i-2}+f(i-1)$ ，所以每个 a_i 在 k 倍的地方的 p 值会产生 $+1$ 的影响。

我们考虑如果对于斐波那契数列的第 i 项我们对它加一个 1 并且继续进行后面的递推的话，那么第 j 项 （$j>i$）的值就是 $\text{fib}_j+\text{fib}_{j-i+1}$.

然后对于每个 a_i , 考虑每个 a_i 对 p_m 的影响，即这个 a_i 会给答案贡献

$$\sum_{k=0}^{ka_i+m \bmod a_i\le m}\text{fib}_{ka_i+m \bmod a_i}$$

暴力计算，时间复杂度 $O\left(\sum_{i=1}^{n}\frac{m}{a_i}\right)$, 可以通过 $13\sim16$ 测试点，获得另外 20 分。

算法肆

仍然考虑矩阵快速幂来优化算法叁的过程。

设 B 为表示 $\text{fib}_{m \bmod ai}$ 的矩阵，那么 $p_{ka_i+m \bmod a_i}$ 可以表示为 BA^{ka_i} ，然后问题就变成了快速求 $\sum_{i=1}^{n}a^i \mod M$。

> *设* $f(n)=\sum_{i=1}^{n}a^i \mod M$
>
> *求和过程用了二分的思想，直观地举例说明：*
>
> $$\begin{cases}a+a^2+a^3+a^4=a+a^2+a^2(a+a^2) & n\text{ is even}\\ a+a^2+a^3+a^4+a^5=a+a^2+a^3+a^3(a+a^2) & n\text{ is odd}\end{cases}$$
>
> *形式化地：*
>
> $$f(n)=\begin{cases}f\left(\frac{n}{2}\right)+n^{\frac{n}{2}}+n^{\frac{n}{2}}f\left(\frac{n}{2}\right) & n\text{ is even}\\ f\left(\left\lfloor\frac{n}{2}\right\rfloor\right)+n^{\left\lceil\frac{n}{2}\right\rceil}+n^{\left\lceil\frac{n}{2}\right\rceil}f\left(\left\lfloor\frac{n}{2}\right\rfloor\right) & n\text{ is odd}\end{cases}$$
>
> *这样，就可以在* $O(log_2^2n)$ *的时间复杂度内解决这一问题了。*

利用这一技巧，总的时间复杂度降为 $O(nlog_2^2n)$.

需要注意的是，矩阵乘法的常数较大，可以通过循环展开优化，使程序更快地运行。

这样就可以获得满分了。

最后，~~强烈推荐《某科学的超电磁炮》~~希望大家喜欢这套题目。