CSP-S 2019 模拟赛题解

foreverpiano

November 5, 2019

1 graph

1.1 满分解法

搜索每一个点是否被选，用一个 for 判断这个点是否可以被选择。可以加剪枝, 如果当前不存在一个点可以选择了就退出搜索。因此搜索的层数不会超过 5 层。如果对最后一层进行搜索进行特判，并且使用 bitset 优化, 复杂度为 $O(\frac{n^4}{\omega})$ 。

2 candy

2.1 满分解法 1

看到随机就应该信仰乱搞，这是一个乱搞做法。给点随机一下，做 200 次以下的过程: 假设我们已经求出了前 i 个点的答案，考虑加入一个点，从 $\binom{4}{3}$ 个三角形中选出最大的面积，并作为第 $i+1$ 的答案。当 n 比较大的时候，对边缘的点特殊处理，即按照 x 排序，取边缘 1000 个点做上述过程。

2.2 满分解法 2

先给点集求一个凸包，暴力枚举凸包上的三个点，复杂度为 $O(K^3)$，其中 K 为凸包点的数量，K 的一个界是 $O(\sqrt{n})$ 。考虑 n 个点的最长上升子序列的期望是 $O(\sqrt{n})$ ，把点的 y 坐标看成数值，x 坐标看成下标，就可以得出上面的界。

3 gcd

3.1 80% 解法

先预处理出所有四元组 (x, l, r, i) 表示左端点在 $[l, r]$ 内, 右端点为 i , $\gcd = x$ 。可以通过线段树维护区间 gcd ，并且二分区间左右端点来完成，复杂度为 $O(n \log^2 n)$ 。

接下来我们固定 x , 并且按照 i 从 n 到 1 处理 $\gcd = x$ 时区间的最大值和数量。对于每一个位置记录 f_i 表示到 i 位置选取的最大区间数，每一次操作取 $[i, n]$ 的选取区间最大的数量来更新 $[l, r]$ 的 DP 值，可以使用线段树区间加来完成。这一部分复杂度为 $O(n \log^2 n)$ 。

3.2 满分解法

预处理四元组的部分，对于同一个 i ，随着左端点往左移动，gcd 为单调的减小，并且都是 val_i 的因数，所以这样的二元组只有 $O(n \log n)$ 组。至于如何预处理，用 i 的四元

组来更新 $i+1$ 的四元组，从后往前拿四元组和 val_i 取 gcd ，同样 gcd 也只会变化 $n\log n$ 次，所以预处理部分的复杂度为 $O(n\log n)$ 。

注意到对于一个固定的 i ，对应的 $[l,r]$ 都是单调的，这意味着可以利用单调性来优化这一过程。考虑维护两个指针 x,y ，分别表示 $<l$ 的第一个位置满足 f_i 最大，在 $[l,r]$ 区间内的最大的 i 。我们让 i 从 1 到 n 扫，对于 x,y 贡献的处理需要维护 $\sum f_x, \sum f_x x$ 。可以 $O(1)$ 时间更新状态，复杂度为 $O(n\log n)$ 。