2020.08.15-2020.08.21 周报

团队训练

比赛时间	比赛名称
2020.08.21	2020 Multi-University Training Contest 6

团队会议

无

个人训练 - nikkukun

专题

无

比赛

2020.08.14 yukicoder contest 261

题目	A	B	C	D	E	F
通过	√	√	√	√	√	
补题						

2020.08.14 Educational Codeforces Round 93 (Rated for Div. 2)

题目	A	B	C	D	E	F	G
通过	√	√	√	√	√	×	
补题						√	√

2020.08.15 AtCoder Beginner Contest 175

题目	A	B	C	D	E	F
通过	√	√	√	√	√	×
补题						√

- C 题变量 typo，WA(-1)，
- D 题空间开少了 RE(-1)，然后忘改语言 RE(-2)，虚空调了十几分钟才发现交错语言了；
- F 题赛后 10min 调出来了，我实在太喜欢赛后过题了。

2020.08.16 Codeforces Global Round 10

题目	A	B	C	D	E	F	G	H	I
通过	√	√	√	√	√	√			

题目	A	B	C	D	E	F	G	H	I
补题									

学习总结

无

个人训练 - qxforever

专题

比赛

2020.08.16 Codeforces Global Round 10

题目	A	B	C	D	E	F	G	H	I
通过	√	√	√	√	√	√			
补题									

学习总结

个人训练 - Potassium

专题

无

比赛

无

学习总结

无

本周推荐

nikkukun

Yukicoder P1172 - Add Recursive Sequence

- **题意**：（方便起见，部分记法与原题不同）$a_0, a_1, \ldots, a_{\infty}$ 是一个 $k \leq 200$ 项常系数齐次线性递推数列，即对 $p \geq k$ 都有 $a_p = \sum _{i=1}^k a_{p-i} c_i$）且所需参数都已给定。现有一个长度为 $n \leq 10^5$ 的序列 $\{ x_n \}$）初始值都为 0，接着进行 q 次操作，每次操作会选定一个区间 $[l, r]$）依次将该区间内对应的值加上 $a_0, a_1, \ldots, a_{r-l}$）求最后序列中每个位置的值模 $10^9 + 7$。
- **题解**：首先考虑如何计算某个位置上 x_i 的值。不妨假设所有区间端点都距离 i 充分远，则 x_i 也可以由它之前的 k 项以 $c_1, c_2, \ldots, c_k$ 为系数递推得到（比较显然，相同递推的和式系数不变），因此可以维护一个 $f_i = x_i$）每次用 $f_{i-k}, f_{i-k+1} \ldots, f_{i-1}$ 推出 x_i）这部分的复杂度是 $O(nk)$ 的。
- 接着考虑区间端点距离 i 并不充分远，使得 x_i 中可能出现并没有递推关系的 $a_0, a_1, \ldots, a_{k-1}$ 的贡献（它们并不能通过递推得到）。我们可以先不将这一部分贡献加入 f_i）而是每次暴力将 i 上 $a_0, a_1, \ldots, a_{k-1}$ 的贡献加入 x_i）然后只在某个区间准备对 x_i 贡献 a_k 这一项时，才给 $f_{i-k}, f_{i-k+1} \ldots, f_{i-1}$ 依次加上 $a_0, a_1, \ldots, a_{k-1}$）按之前提到的方法计算递推部分的贡献。这部分的复杂度是 $O((n + q)k)$ 的。
- 综上，总时间复杂度 $O((n + q)k)$）
- **备注**：需要利用常系数齐次线性递推数列的性质，分开计算与维护 $<k$ 部分的贡献与 $\geq k$ 部分的贡献，还是比较巧妙的。

qxforever

题目名称

- **题意**）
- **题解**）
- **备注**）

Potassium

The Kth Largest Value

- **题意**：给一个有向图，定义 (u,v) 是好的当且仅当 u 可以通过某些路径到达 v）如果 (u,v) 是好的，这一个偶对的权值定义为 $u\oplus v$）q 次询问，每次求第 k 大的好的偶对的权值）$n\le 50000,m\le 200000,q\le 10,T\le 3$）
- **题解**：首先显然可以用 bitset 和拓扑排序求出来所有 u 能够到达的 v 的集合，对每次询问，在 trie 上贪心的往下选，当当前选择 1 后能够到达的点数总和 $\geq k$ 时可以选择 1，否则选择 0。故枚举每个点，求有多少个点满足能够从 i 连过来且值在某区间内。这样通过求 bitset 的前缀和可以求解，但是这样的复杂度是 $O(n^2+n\log n)$ 的，预处理复杂度太大。将 bitset 分成 $block$ 块，把每块看成一个整体求前缀和，这样预处理复杂度 $O(n^2block)$）询问复杂度 $O(q\cdot 2nblock\cdot\log n)$）空间复杂度 $O(n^2block)$）$block$ 取 40 的时候可以非常极限地卡过。
- **备注**：分块 nb

From:
https://wiki.cvbbacm.com/ - **CVBB ACM Team**

Permanent link:
https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:i_dont_know_png:week_summary_16&rev=1598008742

Last update: **2020/08/21 19:19**