

比赛信息

- 日期：2020.5.23
- 比赛地址：[传送门](#)
- 做题情况 lxh(AG) tyx(BCF) gyp(DE)

题解

A - Adjoin the Networks

solved by lxh,tyx

written by lxh

题意

给出一个森林，求一种链接方式，将森林连成一棵树，并让树的直径最短。

数据范围

点数 $0 \leq c \leq 10000$

题解

将两棵树连在一起之后，考虑最小化直径，采取和按秩合并并查集类似的思想，我们就将直径的中点作为根来链接，并对新的直径的大小进行分析，我们不妨记两棵树的直径长度分别为 a, b 且 $a \leq b$ ，则：

若 $b \geq a+3$ ，则新树的直径长度仍为 b ，

若 $b == a+2$ ，当 b 为奇数时，新树直径为 $b+1$ ，当 b 为偶数时，直径长度不改变。

若 $b == a+1$ ，新树直径为 $b+1$ 。

经过以上分析，采取贪心的思想，我们只需要考虑前三大的直径就能保证直径不再扩大。

B - Bell Ringing

solved by tyx

题意

给出一个 n 的全排列的排列，满足相邻两个排列只有两个位置不相同。第一个必须是从 1 到 n 排列。

数据范围

$1 \leq n \leq 8$

题解

先算出 n 的排列，然后让 $n+1$ 在这些排列里面反复移动即可，例如

2 的答案是

```
1 2
2 1
```

3 的答案是

```
1 2 3
1 3 2
3 1 2
3 2 1
2 3 1
2 1 3
```

C - Cryptographer's Conundrum

solved by tyx

题意

给出一个字符串 s 问把它改成 PERPERPER... 这样的循环需要更改几次。

数据范围

$|s| \leq 300$

题解

直接 $O(n)$ 扫一遍即可。

D - Disastrous Downtime

solved by tyx,gyp

written by gyp

题意

现在有 n 个进程要处理，每个进程的长度都是 $1s$ ，开始时间会给出，一个服务器最多同时处理 k 个进程，问最少需要多少个服务器。

数据范围

$n, k \leq 10^5$

题解

找到所有单个毫秒中同时有要处理的进程的最大值即可，由于只需要查询一次，可以直接差分，时间复杂度 $O(n)$

E - Entertainment Box

solved by tyx,gyp

written by lxh

题意

有 n 个区间，选择区间，使得在保证任意时刻不会有超过 k 个区间重合的情况下区间最多。

数据范围

$1 \leq k < n \leq 100000$

题解

这题我们采取贪心的思想，我们按照区间左端点排序加入区间，在已经有 k 个重叠的情况下，如果新加入第 $k+1$ 个区间，显然我们要弹出区间右端点最大的区间，同时在读入新的区间时，将右端点小于当前区间左端点的区间弹出，用平衡树来维护这种关系即可。

F - Floppy Music

solved by tyx,gyp

written by tyx

题意

现在有 $T + 1$ 个位置，要找一个位置使得这个位置满足以下规则：在某几个给定的时间区间里可以以一个位置一秒的速度不停地移动，在时间区间内不能转向，区间之间可以转向，保证区间不会相连，问是否可以找到一个位置。

数据范围

$T \leq 10^4$ 区间个数 $n \leq 100$ 数据组数 $f \leq 10$

题解

一开始把每个位置置成 1，然后每次找到一个位置后根据区间长度 L 检查这个位置往左和往右 L 的位置在不在 $T + 1$ 个位置之中，如果在就把那个位置置成 1（在另一个数组），一共有区间个数个数组，时间复杂度为 $O(Tnf)$

G - Goblin Garden Guards

solved by tyx,gyp

written by lxh

题意

在一个平面给出一些点，再给出一些圆，问有多少点没有被覆盖。（一个坐标上可以有多个点）

数据范围

点数圆数 ≤ 100000 ，坐标 $0 \leq x_i, y_i \leq 10000$ ，圆的半径 $r \leq 100$ 。

题解

本体坐标的范围较大，我们不妨考虑从圆的半径这较小的一维来切入，采取对点每行用一个 $vector$ 来保存，之后对于每个圆，由于只涉及到最多 200 行，对每一行取 $lower_bound$ 和 $upper_bound$ 来

得到圆覆盖的位置，并采取在开始位置 $+1$ ，在结束位置 -1 的方式来标记覆盖的点（因此 $vector$ 里的变量应当是 $pair$ ，最后遍历所有 $vector$ 来统计标记前缀和为 0 的点的数量。

Replay

第一小时 tyx, lxx 先想 A 想出后 lxx 开始写 A gyp 想 tyx 发现 C 题很多人过，于是 A 掉了 C 题， gyp 没有相处 lxx 转而想 E tyx 想出 D 后 gyp 优化了其算法并 A 掉了 D 题。 lxx A 错了几次后发现算法有瑕疵，更改后通过 A

第二小时 gyp 想出 E 但是不会实现，给 lxx 讲解后 lxx 开始写 E 并通过 tyx 和 gyp 想出了 G 给 lxx 讲解后 lxx 开始写 G tyx, gyp 开始看 F 但没有看懂。

第三小时 lxx A 了 G tyx 看懂了 F 并 A 掉了 F gyp 继续想 I 但没有想出。

第四小时：三个人先看了 H lxx 和 gyp 其中 H 没有看懂 tyx 和 gyp 看懂了但没有想出，之后三个人一起看 B 发现是构造题 tyx 先写但没有写出 tyx 想到了构造方法，但花了半个小时左右才写出 A 掉。

第五小时 tyx 看懂了 H 但不会做，三个人开始想 lxx 隐约想到了做法但并不知道细节怎么写。最后没有人 A 题。

总结

- 要仔细看清楚题目要求。
- DP 方面的知识点还需要加强。

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

Permanent link:
<https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:hotpot:nordiccollegiateprogrammingcontest2015&rev=1590746347>

Last update: 2020/05/29 17:59