

## 比赛信息

- 日期：2020.5.2
- 比赛地址：[传送门](#)
- 做题情况  $\square$  [lxh\(BG\)](#)  $\square$  [tyx\(D\)](#)  $\square$  [gyp\(J\)](#)

## 题解

- A - Avaricious Maryanna
  - solved by -, upsolved by tyx,lxh,gyp
  - 题意：给定 $T$ 个询问，每次询问一个 $N$ 位要求输出所有 $N$ 位的数 $x$ 的满足 $x*x$ 的后 $N$ 位等于 $x$ 。(不妨假设 $x$ 为 $i$ 位)。
  - 数据范围  $\square$   $N \leq 500$   $\square$   $T \leq 1000$   $\square$
  - 题解：首先我们不难想到满足条件的1位数分别为0, 1, 5, 6。我们可以想到，满足条件的两位数的个位必定是满足条件的一位数... $N$ 位后面的数字必定是 $N-1$ 位满足条件的数字，由于有这样的性质，我们可以在这4个数的基础上在前面添数字进行验证，由于最多有500位，需要高精度，复杂度应为 $(N*N*4*10)$ ，特别的，我们需要注意，满足条件的还前导0的 $N$ 位数我们也要考虑，例：我们需要从0625推出90625，4位不保存则会少解。然后可以发现，这样到 $N$ 位满足的不会超过4个，且必定有两个为000...1和000...0。然后发现用高精预处理答案会 $T$ (或许写万进制或者亿进制不会?)，其实只要通过高精度的算法把500位的两个解弄出来每次去掉最高位就可以了(无前导0即为解，有则不为)。
- B - Boring Homework
  - solved by lxh
  - 题意：给定 $T$ 个询问，每次给定一个数字从 $1-n$ 的序列，按序列生成BST并在每列只允许出现一个点(代表树上的点)的情况下输出BST要求有左右子树的情况下需要在 $o$ 一边加+号，通过添加-号来维护每列只能有一个点的限制，最小化-号的个数。(具体可以看原题)
  - 数据范围  $\square$   $N \leq 80$   $\square$   $T \leq 2500$   $\square$
  - 题解：经过分析样例和思考之后我们不难发现，对于一个点所处的位置，一定是处在它的权值那一列的，在这种构造情况下，对于一个点 $o$ ，我们需要从它的编号-1-它的左子树这一位置开始插入它的左子树的右子树这么多个-...(后面的减号和空格同理可以推得)，用宽搜和打标记(处理)的方式输出即可。
- D - Detection of Extraterrestrial
  - solved by tyx
  - 题意  $\square$   $T$ 组询问  $(T \leq 200)$   $\square$  每次给出一个字符串  $s(|s| \leq 1000)$   $\square$  问字符串中有循环节且循环了  $k(1 \leq k \leq |s|)$  次的子串最大长度，每个字符串的 $1$ 到 $|s|$ 都要回答。
  - 数据范围  $\square$   $N \leq 1000$   $\square$   $T \leq 200$   $\square$
  - 题解：我们知道在KMP算法中，如果有  $i \bmod (i - \text{next}[i]) = 0$   $\square$  说明这个字符串存在循环节，我们根据这一结论枚举字符串的起点和终点即可，复杂度  $O(n^2T)$   $\square$  本来以为过不了，没想到只跑了120+ms  $\square$
- E - Entertainment
  - solved by -, upsolved by gyp
  - 题意：给定一个人本方发球的胜率和对方发球的胜率，计算这个人一场网球比赛的胜率。网球比赛发一次球，胜者得一分。一个人在一局获胜的条件是至少赢四分且超过对手两分；一个人在一盘获胜条件是赢至少六局且超过对手两局；先赢得三盘的人获得一场的胜利。每局的发球人不变，一盘中的相邻两局发球人交替。一场的相邻两局率先发球者交替。所计算的人率先发球
  - 数据范围  $\square$   $T \leq 10^4$   $\square$
  - 题解：先算出这个人一局中先发球和后发球各自的胜率。然后计算一盘中先发球和后发球各自的胜率。然后计算率先赢得三盘的胜率。计算时，可以先假设每局赛满6球，每盘赛满10局，

每场赛满5盘进行计算。如果一局3:3或一盘5:5，再用级数求和算得这种情况下的获胜几率。

## Replay

## 总结

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

Permanent link: <https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:hotpot:acmgooglecup2011invitationalprogrammingcontest&rev=1588741607> 

Last update: **2020/05/06 13:06**