

codeforces round 659(div2)

=====入坑以来第一次爆零，我裂开了.....

A. Common Prefixes

题意：给 \$n\$ 个数字，要求输出 \$n+1\$ 个字符串，满足，第 \$i+1\$ 个字符串和第 \$i\$ 个字符串有 \$a_i\$ 个相同的地方。

题解：每次前 \$a_i\$ 个字符照上面复制下来，后面所有数，就填上面所有字符的后一位即可。

B Koa and the Beach (Easy Version)&&(hard Version)

题意：一个人要从 0 号沙滩到 \$n+1\$ 号岛屿，中间有 \$n\$ 片海，每片海都有一个深度 \$d_i\$ 还存在潮汐的作用，潮汐在呈 \$[0, 1, 2, \dots, k-1, k, k-1, \dots, 1]\$ 的规律按照时间循环变化，在一个时刻海域的深度为潮汐高度加深度，有一个安全深度 \$l\$ 而且他移动一次需要时间 \$t\$，它可以在任何一个海域停留任何一个时间。问他能否安全的到达目标岛屿？

题解1：对于简单的版本，考虑使用 \$dp\$ 的思想，用 \$dp[i][j]\$ 表示在时间为 \$j\$ 时在第 \$i\$ 个海域的可能性，转移时考虑 \$dp[i][j-1]\$ 与 \$dp[i-1][j-1]\$ 有没有存在一个 1，有就可以转移，时间总量可以开大一点，开 \$2*k*n\$ 最后遍历 \$dp[n][j]\$ 存找是否存在 \$1\$ 即可，至于判断单个状态是否成立，这很简单。

题解2：上面那种比较暴力得做法，明显过不了大样例，所以要想一个 \$O(n)\$ 的想法，其实就是在设计一个无论在那种情况都能占优的策略，可以发现，在降潮的时候走是有优势的，如果这时候到 \$i+1\$ 会有危险，那就等到 \$i+1\$ 没有危险时再走，长时间呆在原地，潮水只会往下跌，所以只会越来越“安全”，所以只要找可以等到涨潮的点即可，即满足 \$k+d_i \leq l\$ 的点，在这个点等到潮水涨到最高，然后再开始继续走，当然，一开始要判断不成立，两种情况，一种是已经涨潮，而对面已经过不去了，还有就是存在一片海域 \$d_i > l\$，那么也一定不成立。时间复杂度 \$O(N)\$

C. String Transformation 1

题意：给两串字符串，每次可以经行一个操作，人选 \$a\$ 字符串中一个字母 \$s\$ 可以选 \$a\$ 中人一多个 \$s\$ 将其变为比它字典序大的一个字母，问最少多少次操作可以实现把 \$a\$ 字符串变成 \$b\$

题解：采取从低位到高位的策略，这样可以尽量顾及到，连续变化削减次数的情况，即每次将所有 \$a\$ 中的字符 \$i\$ 都变成字符 \$j\$ 直到没得变为止。因为变得顺序是从小字母到大字母，所以不会出现，变了之后 \$b\$ 中字母反而小于 \$a\$ 中字母的情况。可以采取符号维护 \$f[i][j]\$ 解决，\$f\$ 表示 \$a\$ 为 \$i\$, \$b\$ 为 \$j\$ 是否存在。

D GameGame

题意：给 \$n\$ 个数，两个人进行博弈，一次去一个数，对自己现有的数取异或，问最后的输赢情况。

题解：看到异或不难想到一位一位的看，从高到低的考虑每一位，如果这一位是偶数个 1，没有意义，两

个人都会拿到相同个数个1，最后这一位不会产生差别，如果这一位是奇数个1。分三种情况。

- 数量 p 满足 $p \% 4 == 1$ 时。这时，先手必胜，先手先取一个1，然后后手取啥先手就去啥，最后一定先手会拿到奇数个1，后手偶数个1，最后结果先手这一位是1，后手为0。
- $p \% 4 == 3$
 - n 为奇数时，先手必输，无论先手选什么，后手跟先手选一样的，直到不能选为止，最后一定有先手有偶数个\$1\$，后手有奇数个\$1\$，
 - n 为偶数时，先手必胜，先手现选一个0，就让后手的转台转换为上一种情况，这是必输的，所以先手必赢

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

Permanent link:
https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:manespace:codeforces_round_659_div2&rev=1596183526

Last update: 2020/07/31 16:18