

codeforces round 660 div 2

链接：<https://codeforces.com/contest/1388>

拉胯 我疯狂的拉胯
甚至b忘记交了

A

题意：定义两个质数的积是“nearly prime”的。求 n 能否拆成四个不同正整数之和，其中三个是“nearly prime”的，并求方案。

解：相当的显然，最小的三个“nearly prime”分别为 6, 10, 14。那么只要 $n > 30$ 就可以了。注意一下如果是类似 [6, 6, 10, 14]，因为下一个“nearly prime”是 15，整理成 [5, 6, 10, 15] 即可。10, 14 也类似操作

B

题意：例如一个数 729，将其每一位都转二进制，并拼起来，会得到 111101001。给定 n ， n 位数中，按前述操作完，并去掉最后 n 位，最大的数（这个数显然可以由若干个 n 位数转换得到）。求前述（转换二进制并去掉最后 n 位后最大）的这些 n 位数中最小的。

解：题意很绕，建议看原题。做法是：先将 n 个 9 的十进制转二进制，然后在最后 n 位改 1000，从右向左改。最后转回 10 进制输出。即，它应该是“若干个 9 后接若干个 8”的结构。
证明它是解如下：首先每一位只能是 9 或 8，否则转换操作后长度不是最长，去掉 n 位后就必定不是最大。然后因为最后 n 位被去除了，所以它们尽量是 8（因为要最小）。剩下的，前面都填 9（因为要最大）

C

题意：一棵有根树，1 为根。每个点 i 有 p_i 个人作为目标，每个人有值 $\{-1, 1\}$ 。过程中，每个人都要从 1 走到他自己所在的点，路上他的权值可能从 1 变为 -1（不可能反过来）。在每个点记录经过该点的人的权值和 h_i ，求 h_i 是否是合法的。

解：暂略。来不及写完了，是个从下往上的倒退。

D

题意：给定 n 个 a_i, b_i 初始 $ans = 0$ ，每次操作：选定 i ，先 $ans += a_i$ ，再 $a_{b_i} += a_i$ 如果 $b_i \neq -1$
题目保证序列 b_i, b_{b_i}, \dots 必定以 -1 结尾（即不会成环）。求合法的操作序列使得 ans 最大。注意 a_i 不保证是正数。

解：显然这构成一棵树。对 i 操作后还会使得 a_i 加到它的父亲上。树形 dp，对于 i 在每个子树都维护了最大的 a_{root} 后，将非负的 a_{root} 加到 a_i 最后统计 ans 对于输出方案，则先输出非负子树的方案，再输出 i 再输出负的子树，保证负的子树的根不会被加到父亲上。

注意 \$ans\$ 和 \$a_i\$ 可能会爆 int ，同时注意可能有多颗子树。

E

还没看

From:

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

Permanent link:

https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:no_morning_training:fayuanyu:cf_r660d2

Last update: **2020/07/31 16:45**

