

codeforces 641 (div2)

A

题意：给定一个数 n 经行 k 次操作，每次将 n 最小的非1约数加在 n 上。

解：对于奇数加上最小约数后变偶数(素数当然也是奇数，所以同理)，最小约数为2，而对于偶数最小约束一直是2。需要做的就是 $O(\sqrt{n})$ 的时间复杂度下处理出最小非1约数即可。

B

题意：大致意思是要你从一段序列中选择一段不一定要连续的子列，使其满足上升且，下标满足 $i < j$ 且 $i | j$

解：明显是 dp 转移也不是很难，从1到 n 遍历一遍，然后再分别枚举倍数，复杂度 $O(n\sqrt{n})$

借着代码解释一下

```
for(int i=1;i<=n;i++)
{
    for(int j=1;i*j<=n;j++)
    {
        if(a[i]<a[i*j]) dp[i*j]=max(dp[i*j],dp[i]+1),ans=max(ans,dp[i*j]);
        else continue;
    }
}
```

C

题意：大致意思是，给定一段序列，然后任取两个数取其最小公倍数，然后放入新序列，问将所有情况取遍后，那整个序列的最大公约数。

解：数论.....脑阔疼.....

官方版：听说巨佬推出公式，反正我是不会推，对于序列 $\{a_i\} (1 \leq i \leq n)$ ，有一个公式为 $\gcd(\{a_i, a_j\}) = \text{lcm}\{a_i, \gcd(a_i)\} (1 \leq j \leq n \text{ 且 } j \neq i)$ 则这里 a_i 取遍所有可能情况，这样我们就可以，预处理出 \gcd 的前缀，然后再求 \gcd

暴力版：太难了，我只会暴力.....记过一系列简单计算，不难发现，结果即为，将所有数的素因子集合起来，筛选出在序列中出现过至少 $n-1$ 次的素因子(否则答案中一定不存在)，对于所有满足条件的素因子 i 遍历一遍序列求出序列所有数中素因数分解式中 i 的指数，取第二大的，则答案的素因数分解中一定存在这个项。接下来线性筛暴力筛出所有素数然后暴力循环找即可。

D

题意：给一段序列，给定一个数 k 操作为：每次取一段区间，求出区间中位数，将区间所有数变成那个数，问能不能全刷成给定的数。

解：找规律的题，找不出来就罚坐(真毒瘤)，大致我们要判定这样一件事情，只有一个数，判定是不是那

Last
update: 2020/05/16 2020-2021:teams:manespace:codeforces_642_div2 https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:manespace:codeforces_642_div2
18:00

个数, 给许多数, 若序列中都不存在那个数, 明显不满足, 若满足, 则是否存在三个数(两个数就判定两个数), 满足有两个数大于等于k

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

Permanent link:
https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:manespace:codeforces_642_div2 

Last update: **2020/05/16 18:00**