

## 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，而对于偶数最小约数一直是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;
    }
}
```

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

Permanent link:  
[https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:manespace:codeforces\\_642\\_div2&rev=1589621740](https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:manespace:codeforces_642_div2&rev=1589621740)

Last update: 2020/05/16 17:35

