

# Codeforces Round #699 (Div. 2)

[比赛链接](#)

## E. Sorting Books

### 题意

给定一个长度为  $n$  的序列  $a$  每次操作可以任取一个位置并将该位置的数移到序列末尾。

问最少需要多少次操作才能使序列中所有大小相同的数相邻。

### 题解

显然只需要在序列中选定一些位置，然后按特定顺序将他们移除就可以完成任务。定义  $\text{dp}(i)$  表示不选择序列  $[i, n]$  中的位置个数的最大值。

维护每个值  $v$  的最靠左出现的位置  $l_v$  最靠右出现的位置  $r_v$  动态维护  $[i, n]$  中出现的次数  $c_v$

如果保留位置  $i$  当  $i = l_{a_i}$  时，需要移除  $[i, r_{a_i}]$  中所有不等于  $a_i$  的位置，即  $\text{dp}(i) \leftarrow c_{a_i} + \text{dp}(r_{a_i} + 1)$

当  $i \neq l_{a_i}$  时，若不留  $l_{a_i}$  的位置，则  $l_{a_i}$  的位置的数会被加到位置  $n$  后面。

为了使得  $l_{a_i}$  的位置的数与  $i$  位置的数所在段相邻，需要移除  $[i, n]$  中所有不等于  $a_i$  的位置，于是  $\text{dp}(i) \leftarrow c_{a_i}$

若保留  $l_{a_i}$  的位置，则该情况会在计算  $\text{dp}(l_{a_i})$  时考虑，此时不考虑。

如果不保留位置  $i$  则  $\text{dp}(i) \leftarrow \text{dp}(i+1)$  最终答案为  $n - \text{dp}(1)$  时间复杂度  $O(n)$

```
const int MAXN=5e5+5;
int a[MAXN], dp[MAXN], l[MAXN], r[MAXN], c[MAXN];
int main()
{
    int n=read_int();
    _rep(i, 1, n)a[i]=read_int();
    for(int i=n; i; i--) {
        l[a[i]]=i;
        if(!r[a[i]]) r[a[i]]=i;
    }
    for(int i=n; i; i--) {
        c[a[i]]++;
        dp[i]=dp[i+1];
    }
}
```

```
    if(i==l[a[i]])dp[i]=max(dp[i],dp[r[a[i]]+1]+c[a[i]]);
    else dp[i]=max(dp[i],c[a[i]]);
}
enter(n-dp[1]);
return 0;
}
```

## F. AB Tree

### 题意

### 题解

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

Permanent link:  
[https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:legal\\_string:jxm2001:contest:cf\\_699\\_div\\_2&rev=1612750005](https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:legal_string:jxm2001:contest:cf_699_div_2&rev=1612750005)

Last update: 2021/02/08 10:06

