

A	B	C	D	E	F
+	+	+	+	+	0

rank:93

## A

- 题意: $n$ 个数,问能否选 $x$ 个数使他们的和是奇数( $x \leq n \leq 1000$ )
- 题解:不带脑子WA子两发。 $n$ 太小了,直接暴力枚举即可。

## B

- 题意: $01$ 串,每次操作将一个字符翻转,问将串变为 $01$ 或 $10$ 出现次数 $\leq 1$ 的串最少要操作几次。
- 题解:维护前后缀和,考虑先 $1$ 后 $0$ 或先 $0$ 后 $1$ 然后扫一遍即可。

## C

- 题意:给一棵树,两人一人一次操作,每次可以移除度 $\leq 1$ 的节点,谁先移除掉给定点就获胜,问先手赢还是输。
- 题解:如果给定点直接可以移除,那么显然赢。否则可以发现每次都可以只删其它点,直到只剩下 $2$ 个点,因此先手胜等价于节点数是偶数。

## D

- 题意:交互题。给定一个长度为 $n$ 的序列 $a$ 和 $k$ 个两两没有交集的非空下标集合,每个集合有一个权值,等于除了该集合中下标对应的序列 $a$ 中的元素以外 $a$ 中其它元素的最大值。给出这 $k$ 个集合,你可以询问不超过 $12$ 次,每次询问 $a$ 中任意个指定元素的最大值。最后要给出 $k$ 个集合对应的权值( $2 \leq n \leq 1000, 1 \leq k \leq n$ )
- 题解:可以发现只有一个集合包含了全局最大值的下标(如果有多个最大值,随意挑一个当作全局最大值),那么其它集合对应的权值就是这个全局最大值。我们只需要二分出这个全局最大值的下标,然后询问一下对应的集合的最大值。正好全局 $(1)$ +二分 $(10)$ +最后询问一次 $(1)$ 正好 $12$ 次。需要注意不保证所有下标都出现在集合中,因此有可能全局最大值对应的下标不属于任何一个集合,这个时候所有权值就都是这个全局最大值了。

## E

- 题意:给出一棵树,每个点可能是 $0$ 或 $1$ ,同时每个点需要最后变到 $0$ 或 $1$ 。每个点有一个权值 $a_i$ 每次可以选择一个点将其子树中任意 $k$ 个点的权值任意分配,耗费 $a_i \times k$ 求最小花费,或判断无解。
- 题解:如果原本的 $01$ 数量和要求的数量不同,显然无解。否则考虑每个需要置换的点一定是在它权值最小的祖先处置换,如果某个点有奇数个需要置换的,则继续向上传即可dfs+set模拟这个过程即可。

## F

- 题意:给定两个长度为 $n$ 的字符串 $s$ 和 $t$ 。每次可以选择一个区间将 $s[l, l+1 \dots r]$ 变成 $s[r, l, l+1 \dots r-1]$ 。求最少需要几步可以将 $s$ 变成 $t$ 或判断无解。 $(1 \leq n \leq 2000)$
- 题解:这个操作等价于将某个字符移动到它左边的任意位置,为方便操作,我们将两个字符串翻转,这个操作变为将某个字符移动到它右边的任意位置。设 $f_{i,j}$ 为 $s[1..i]$ 和 $t[1..j]$ 匹配需要的最少操作数,这里 $i \geq j$ 。多余的部分移动到后面去了。转移分为三种:1.如果 $s[i]=t[j]$ 那么这一位可以直接匹配,即 $f_{i,j}=f_{i-1,j-1}$ 。2.将 $s[i]$ 移动到后面某一位,操作数我们在第三步转移算,即 $f_{i,j}=f_{i-1,j}$ 。3.如果 $t[j]$ 在 $s[1..i]$ 的出现次数 $\geq$ 在 $t[1..j]$ 的出现次数,那么相当于有多余的 $t[j]$ 移动到后面来了,因此此时可以使用这个多余的 $t[j]$ 。此时进行了一次操作,因此 $f_{i,j}=f_{i,j-1}$ 。三种转移都取 $\min$ 即可。

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

Permanent link:  
[https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:farmer\\_john:jjleo:codeforces\\_round\\_646\\_div\\_2&rev=1591411489](https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:farmer_john:jjleo:codeforces_round_646_div_2&rev=1591411489)

Last update: 2020/06/06 10:44