

2020/07/18 – 2020/07/24周报□week11□

本周推荐

by iuiou

- 题源□<https://codeforces.com/contest/1385/problem/G>
- 题意：给定两排数，每排都有 n 个数，每次操作可以交换一列的两个数，问能否存在一个最少的交换方案，使操作之后每一行都是 1 到 n 的一个排列。
- 知识点□dfs
- 题解：这题有点难想，大致是一个二分图的问题。首先遍历两行数，如果一个数的出现次数超过了2次，那么肯定不成立。如果所有数出现次数都是两次，那么一定有一种方案满足。现定四个数组 $r1[n], r2[n], b1[n], b2[n], b$ 数组用于存放列数□ r 数组用于存放行数，如果对于一个数 $i, b_{1i} = b_{2i}$ ，则不考虑这个点，因为肯定不会动这个点的。接下来对点染色，如果 $b_{1i} \neq b_{2i}$ ，考虑 r 数组，如果 $r_{1i} \neq r_{2i}$ ，则在 b_{1i} 和 b_{2i} 之间连一条权为0的边表示，两点染的色必须相同。反之，连一条权为1的边，表示两个点颜色相反，最后从每个点开始dfs经行染色即可，注意每次比对将开头的那个点染成1还是0，最后的结果会最优(即最少)。

团队训练

2020.7.18 [牛客多校第三场](#)

2020.7.20 [牛客多校第四场](#)

2020.7.23 [codeforces team prattice](#)

范泽恒

专题

- 无

比赛

- [codeforces round 656\(div3\)](#)
- [codeforces round 657\(div2\)](#)
- [codeforces round 658\(div2\)](#)

题目

- 题源 <https://codeforces.com/contest/1385/problem/G>
- 题意：给定两排数，每排都有 n 个数，每次操作可以交换一列的两个数，问是否存在一个最少的交换方案，使操作之后每一行都是 1 到 n 的一个排列。
- 知识点 `dfs`
- 题解：这题有点难想，大致是一个二分图的问题。首先遍历两行数，如果一个数的出现次数超过了 2 次，那么肯定不成立。如果所有数出现次数都是两次，那么一定有一种方案满足。现定四个数组 $r1[n], r2[n], b1[n], b2[n]$ ， b 数组用于存放列数， r 数组用于存放行数，如果对于一个数 i ， $b_{1i} = b_{2i}$ ，则不考虑这个点，因为肯定不会动这个点的。接下来对点染色，如果 $b_{1i} \neq b_{2i}$ ，考虑 r 数组，如果 $r_{1i} \neq r_{2i}$ ，则在 b_{1i} 和 b_{2i} 之间连一条权为 0 的边表示，两点染的色必须相同。反之，连一条权为 1 的边，表示两个点颜色相反，最后从每个点开始 `dfs` 进行染色即可，注意每次比对将开头的那个点染成 1 还是 0 ，最后的结果会最优（即最少）。

恭天祥

专题

- 无

比赛

- [cf 657 div.2](#)

题目

- 牛客第三次C题：
 1. 题意：给出 20 个点，这 20 个点构成了一只手，通过给定的信息判断这只手是左手还是右手
 2. 题解：这个题很新颖，导致很多之前做题的思路都用不上，开始做题时就死磕，想面积，边长…… 都没想到点上 反而浪费了挺多时间，我觉着这个题重再观察，并且读懂题意，我读题时忘记了手掌面积不变，直接导致了队友懵逼了半小时，这个题的具体解决思路也很简单，大拇指外侧的线长 6 ，手掌底部的线长 9 ，小拇指外侧的线长 8 。判断连续的三根或者两根线是否满足这个规律就行了，由于是浮点数，这题需要控制精度 $\epsilon = 1e-5$ 或更小都行

刘怀远

专题

- 无

比赛

- [Codeforces Round 656 \(Div. 3\)](#)

题目

- cf 656 Problem-D:a-Good String
- 链接：<https://codeforc.es/contest/1385/problem/D>
 1. 题意：多组数据，给定长为 n 的字符串（ n 为2的整数次方），输出将其变成“a - good string”最少需要改动的字符数。以 ch 代指字母，“ ch - good string”定义为：（1）长为1且字符为 ch ；（2）长大于1，且二分后左边字符全为 ch ，右边为“ $ch+1$ - good string”；（3）长大于1，且二分后右边字符全为 ch ，左边为“ $ch+1$ - good string”。
 2. 题解：经典二分dfs，维护最小代价即可。

为什么这破题当初能卡那么久

```
#include<bits/stdc++.h>
using namespace std;

typedef long long ll;
const int maxn = 2e5 + 5;
const double pi = acos(-1);
const int mod = 998244353;

int t, n;
char s[131074];

int cal(int l, int r, int c){
    int cnt = 0;
    for(int i = l; i <= r; ++i) if(s[i] != c) ++cnt;
    return cnt;
}

int dfs(int l, int r, char c){
    if(l == r) return s[l] != c;
    int res = 3e7;
    int mid = (l + r) >> 1;
    res = min(res, dfs(l, mid, c + 1) + cal(mid + 1, r, c));
    res = min(res, dfs(mid + 1, r, c + 1) + cal(l, mid, c));
    return res;
}

int main(){
    scanf("%d", &t);
    while(t--){
        scanf("%d\n", &n);
        scanf("%s", s + 1);
        printf("%d\n", dfs(1, n, 'a'));
    }
}
```

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

Permanent link:
https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:manespace:2020_07_18-2020_07_24%E5%91%A8%E6%8A%A5_week11 

Last update: **2020/07/31 10:33**