牛客多校第八场

想起高中时看的一本大黑砖：

《我怎样爆零□HowDoIGetZero□□

所以预计本周周报将陷入无题可写的困境。

I

比赛时以为是二分图的匹配用匈牙利算法做了，搞了半天搞出来了但是复杂度太高了过不去。

参考题解的答案是：总点数-#□k个点□ k-1条边的连通分量）

```
#include<stdio.h>
#include<string.h>

#include<vector>
#include<algorithm>

using namespace std;

bool vis[200020];
int F[200020];

void init()
{
    memset(F,-1,sizeof(F));
    memset(vis,0,sizeof(vis));
}

int find(int x)
{
    if(F[x]==-1)
    {
        return x;
    }
    return F[x]=find(F[x]);//改了根
}

void bing(int x,int y)
{
    int t1=find(x);//找到x的根
    int t2=find(y);//找到y的根
    if(t1==t2)
    {
        vis[t1]=1;//同一个根，该根变1
        return;
```

```
    }
    F[t1]=t2;//如果x和y的根不同，修改t1的根t2
    if(vis[t1])
    {
        vis[t2]=1;
    }
}

int main()
{
    int t;
    scanf("%d",&t);
    int k=0;
    while(t--)
    {
        k++;
        init();
        int n;
        scanf("%d",&n);
        vector<int> a(n),b(n);
        vector<int> val;
        int i;
        for(i=0;i<n;i++)
        {
            scanf("%d%d",&a[i],&b[i]);
            val.push_back(a[i]);
            val.push_back(b[i]);
        }
        sort(val.begin(),val.end());
        int tot=unique(val.begin(),val.end())-val.begin();
        val.resize(tot);
        for(i=0;i<n;i++)
        {
            a[i]=lower_bound(val.begin(),val.end(),a[i])-val.begin();
            b[i]=lower_bound(val.begin(),val.end(),b[i])-val.begin();
            bing(a[i],b[i]);
        }
        int ans=tot;
        for(i=0;i<tot;i++)//遍历c中不同的点
        {
            if(F[i]==-1&&!vis[i])
            {
                ans--;
            }
        }
        printf("Case #%d: %d\n",k,ans);
    }
}
```

G

看代码觉得这个题明明没有描述的那么难啊……

```
#include<stdio.h>
#include<string.h>

int a[300][5];
char s[100];

char check(int x,int y,int z)
{
    if((x==y&&x==z&&y==z)||(x!=y&&x!=z&&y!=z)||x==-1||y==-1||z==-1)
    {
        return 1;
    }
    return 0;
}

int main()
{
    int t,n,top;
    scanf("%d",&t);
    int kk=0;
    while(t--)
    {
        kk++;
        scanf("%d",&n);
        int i;
        for(i=1;i<=n;i++)
        {
            scanf("%s",s);
            top=0;
            int j;
            for(j=0;j<strlen(s);j++)
            {
                if(s[j]=='[')
                {
                    if(s[j+1]=='*')
                    {
                        a[i][top++]=-1;
                    }
                    else
                    {
                        a[i][top++]=s[j+2]-'a';
                    }
                }
            }
        }
```

```
        char flag = 0;
        printf("Case #%d: ",kk);
        for(i=1;i<=n;i++)
        {
            if(flag)
            {
                break;
            }
            int j;
            for(j=i+1;j<=n;j++)
            {
                if(flag)
                {
                    break;
                }
                int k;
                for(k=j+1;k<=n;k++)
                {
                    if(flag)
                    {
                        break;
                    }
                    char hv=1;
                    int d;
                    for(d=0;d<4;d++)
                    {
                        if(!check(a[i][d],a[j][d],a[k][d]))
                        {
                            hv=0;
                            break;
                        }
                    }
                    if(hv)
                    {
                        printf("%d %d %d\n",i,j,k);
                        flag=1;
                        break;
                    }
                }
            }
        }
        if(!flag)
        {
            puts("-1");
        }
    }
    return 0;
}
```