

2020/07/11 -- 2020/07/17 周报

团队训练

[2020牛客暑期多校第一场](#)

[2020牛客暑期多校第二场](#)

Marvolo

专题

动态点分治

比赛

[Atcoder Alsing Programming Contest 2020](#)

题目

[\[ZJOI2007 捉迷藏\]](#)

Kevin

专题

图论杂项

比赛

无

题目

见本周推荐

TownYan

专题

无

比赛

[Atcoder Alsing Programming Contest 2020](#)

题目

见本周推荐

本周推荐

Marvolo

[\[ZJOI2007 捉迷藏\]](#)

题目大意是给一棵树,然后每个节点会进行黑白染色,求某个时刻,树上最远的两个黑点的距离.

算是动态点分入门题,对于每个点维护两个可删堆来统计答案.每一次修改点的颜色的时候,修改一下点分树中对应的链上每个点的信息就行了.

可删堆的实现比较有意思,是用两个priority_queue封装在一起,一个记录所有元素,一个记录删除了的元素来实现.

Kevin

[\[BZOJ 2115 Xor\]](#)

题意

无向连通图 N 点 M 边, 范围 $1e5$. 求从 1 走到 N 的最大路径权值异或和 (可以不是简单路径)

题解

主要考虑环的影响 (如果无环则 1 到 N 路径唯一)。实际上所有可能路径的异或和可以等于任意一条 1 到 N 路径的异或和再异或上某些环的异或和 (相当于选择岔路)。如果重边和自环也视为环, 这个结论也成立。因此通过无向图 tarjan 或者直接 dfs 得到每个环的异或和, 求异或线性基, 再以随便一条路径异或和 x 为基础, 从高位逐个考虑线性基要不要异或上去, 得到答案。复杂度 $O(kN+M)$ k 是线性基的常数。

代码

```
#include <cstdio>
#define ONE 1ull
```

```
#define builtin_log2(x) (63-__builtin_clzll(x))

using ULL = unsigned long long;

const int MV(5e4+7), ME(1e5+7), MB(61);

struct Edge
{
    ULL d;
    int next, v;
} ed[2 * ME];
int head[MV], tot;
#define edd(_u, _v, _d) ed[++tot].next=head[_u], ed[tot].v=_v, ed[tot].d=_d, head[_u]=tot

bool vis[MV];
ULL dis[MV];
ULL base[66];

void elimnt(ULL v)
{
    if (v)
        for (int i=builtin_log2(v); i>=0; --i)
        {
            if (v & (ONE << i))
            {
                if (base[i])
                    v ^= base[i];
                else
                {
                    base[i] = v;
                    break;
                }
            }
        }
}

ULL get_max(ULL path_xor)
{
    for (int i=MB; i>=0; --i)
        if ((path_xor ^ base[i]) > path_xor)
            path_xor ^= base[i];

    return path_xor;
}

void dfs(const int u)
{
    for (int i=head[u]; i; i=ed[i].next)
    {
```

```
        const int v = ed[i].v;
        if (vis[v])
            elimnt(dis[u] ^ dis[v] ^ ed[i].d);
        else
            vis[v] = true, dis[v] = dis[u] ^ ed[i].d, dfs(v);
    }
}

int main()
{
    int V, E, u, v;
    ULL d;
    scanf("%d %d", &V, &E);
    while (E--)
    {
        scanf("%d %d %llu", &u, &v, &d);
        edd(u, v, d);
        edd(v, u, d);
    }

    vis[1] = true, dfs(1);
    printf("%llu", get_max(dis[V]));

    return 0;
}
```

TownYan

Atcoder::Aising Programming Contest 2020-F

https://atcoder.jp/contests/aising2020/tasks/aising2020_f

题意

求解一个过程看起来简单但繁琐的计数问题

题解

分输入的奇偶分别插值，得到通项

吐槽

插值是真的尝试了，分奇偶是真的第五层

From:

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

Permanent link:

https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:t1e233:week_1_2020_7_11-2020_7_17 

Last update: **2020/07/17 18:13**