

并查集

人员

洪晨栋、洪晨棋、陶汇笙、崔宸赫、于家瑞 到课, 郭栩睿 线上

上周作业检查

上周作业链接: <https://cppoj.kids123code.com/contest/791>

2025-0913周六10:30 (综合练习)									
<div>🔄 刷新</div>									
#	用户名	姓名	编程分	时间	A	B	C	D	E
1	guoxurui	郭栩睿	400	1003	100	100	100	100	
2	hongchendong	洪晨栋	400	1004	100	100	100	100	
3	hongchenqi	洪晨棋	400	1011	100	100	100	100	
4	yuxiaolong	于青龙	385	700	100	100	100	85	
5	yujiarui	于家瑞	300	697	100	100	100		
6	taohuishheng	陶汇笙	300	699	100	100	100	0	
7	cuichenhe	崔宸赫	100	349	100				

本周作业

<https://cppoj.kids123code.com/contest/884> (课上讲了 A ~ D 这些题, 课后作业是 E 题)

课堂表现

今天课上讲了并查集的内容, 并查集的代码非常简短, 但是思想非常重要, 同学们课下要好好再复习一下 A B C 三道题。

课堂内容

Reorder Cards (上周作业)

把所有 用到的行 和 用到的列 用两个 set 存起来, 最后遍历这两个 set, 就是 剩下的行 和 剩下的列

```
#include <bits/stdc++.h>

using namespace std;

const int maxn = 1e5 + 5;
struct node {
    int x, y, id;
} w[maxn];

int main()
{
    int r, c, n; cin >> r >> c >> n;
```

```
for (int i = 1; i <= n; ++i) cin >> w[i].x >> w[i].y;

set<int> s1, s2;
for (int i = 1; i <= n; ++i) s1.insert(w[i].x), s2.insert(w[i].y);

map<int, int> mp1, mp2;
int id1 = 0, id2 = 0;
for (int i : s1) mp1[i] = ++id1;
for (int i : s2) mp2[i] = ++id2;

for (int i = 1; i <= n; ++i) {
    cout << mp1[w[i].x] << " " << mp2[w[i].y] << endl;
}
return 0;
}
```

并查集: 可以约用 $O(1)$ 的时间复杂度 合并两个集合/判断两个数是否在相同集合

【模板】并查集

并查集 模板题

```
#include <bits/stdc++.h>

using namespace std;

const int maxn = 2e5 + 5;
int f[maxn];

int fFind(int x) {
    if (f[x] != x) f[x] = fFind(f[x]);
    return f[x];
}

int main()
{
    int n, m; cin >> n >> m;
    for (int i = 1; i <= n; ++i) f[i] = i;

    while (m -- ) {
        int op, x, y; cin >> op >> x >> y;
        if (op == 1) {
            int fx = fFind(x), fy = fFind(y);
            if (fx != fy) f[fx] = fy;
        } else {
            if (fFind(x) == fFind(y)) cout << "Y" << endl;
            else cout << "N" << endl;
        }
    }
}
```

```
    return 0;
}
```

亲戚

并查集 模板题

```
#include <bits/stdc++.h>

using namespace std;

const int maxn = 5000 + 5;
int f[maxn];

int fFind(int x) {
    if (f[x] != x) f[x] = fFind(f[x]);
    return f[x];
}

int main()
{
    int n, m, p; cin >> n >> m >> p;
    for (int i = 1; i <= n; ++i) f[i] = i;

    while (m -- ) {
        int x, y; cin >> x >> y;
        int fx = fFind(x), fy = fFind(y);
        if (fx != fy) f[fx] = fy;
    }

    while (p -- ) {
        int x, y; cin >> x >> y;
        if (fFind(x) == fFind(y)) cout << "Yes" << endl;
        else cout << "No" << endl;
    }
    return 0;
}
```

村村通

一共 n 个城市, 全合并起来需要 $n-1$ 次

可以设 $\text{cnt} = n-1$, 之后每合并一次就让 $\text{cnt}--$ 就可以了

```
#include <bits/stdc++.h>

using namespace std;

const int maxn = 1000 + 5;
```

```

int f[maxn];

int fFind(int x) {
    if (f[x] != x) f[x] = fFind(f[x]);
    return f[x];
}

int n, m;

void solve() {
    for (int i = 1; i <= n; ++i) f[i] = i;
    int cnt = n - 1;
    while (m -- ) {
        int x, y; cin >> x >> y;
        int fx = fFind(x), fy = fFind(y);
        if (fx != fy) { f[fx] = fy; --cnt; }
    }
    cout << cnt << endl;
}

int main()
{
    while (true) {
        cin >> n;
        if (n == 0) break;
        cin >> m;
        solve();
    }
    return 0;
}

```

素数个数

埃氏筛 模板, 当 $f[i] == \text{true}$ 时, 说明 i 是合数

原理: 所有质数的倍数都是合数

```

#include <bits/stdc++.h>

using namespace std;

const int maxn = 1e8 + 5;
bool f[maxn];

int main()
{
    int n; cin >> n;
    for (int i = 2; i <= n; ++i) {
        if (f[i]) continue;
        for (int j = i+i; j <= n; j += i) f[j] = true;
    }
}

```

```
int res = 0;
for (int i = 2; i <= n; ++i) {
    if (!f[i]) ++res;
}
cout << res << endl;
return 0;
}
```

集合

先用 埃氏筛 找出所有的质数, 然后枚举比 p 大的质数 i , 找到 $a \sim b$ 中所有 p 的倍数, 将他们利用并查集合并。

此题不提供具体代码, 要求同学们课下独立完成