二分图与匹配

2020年1月21日

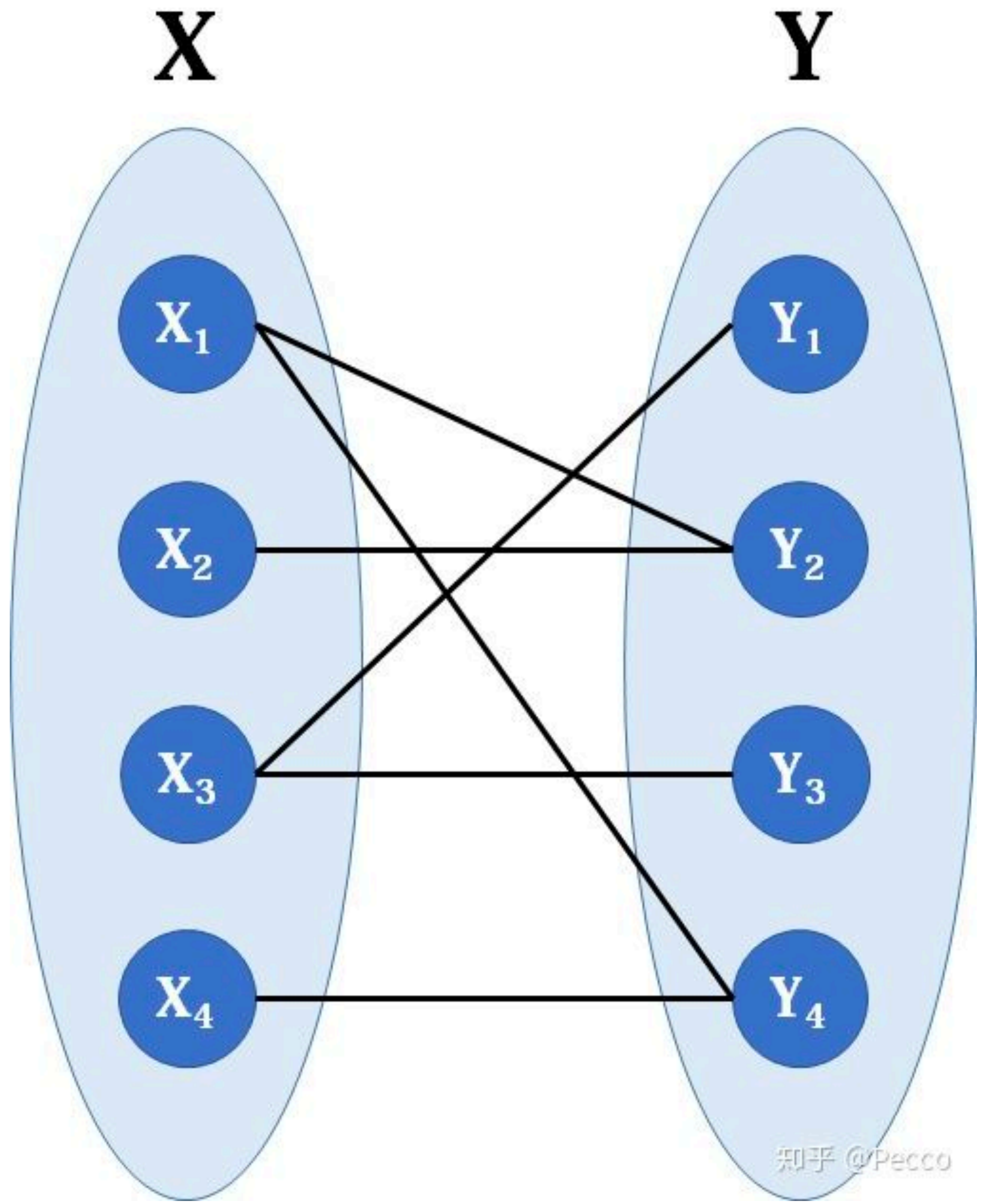
黄哲威 hzwer
北京大学16级计算机科学

课程安排 4

二分图染色

匈牙利算法

二分图



基础概念

二分图: 若对于图 G = (V, E) 存在一个 V 的划分 (A, B) 使得任意一条边的两个端点不属于同一个集合，则 G 是一个二分图

匹配: 在图 G = (V, E) 中，边集 $E' \subseteq E$ 被称为 G 的一个匹配当且仅当对 于 V 中的每个点，E' 中与其关联的边不超过一条

最大匹配: 边数最多的匹配

二分图 <=> 图中没有奇环

二分图染色

二分图的一种等价的说法是，能把图中结点染成黑白两色，使得每条边的两个端点颜色不同。

随便选一个点，染成白色，然后把它相邻的点染成黑色，再把这些点相邻的点染成黑色，以此类推。

用 dfs 或者 bfs 实现这个过程。

如果染色过程中有矛盾，说明这个图不是二分图。

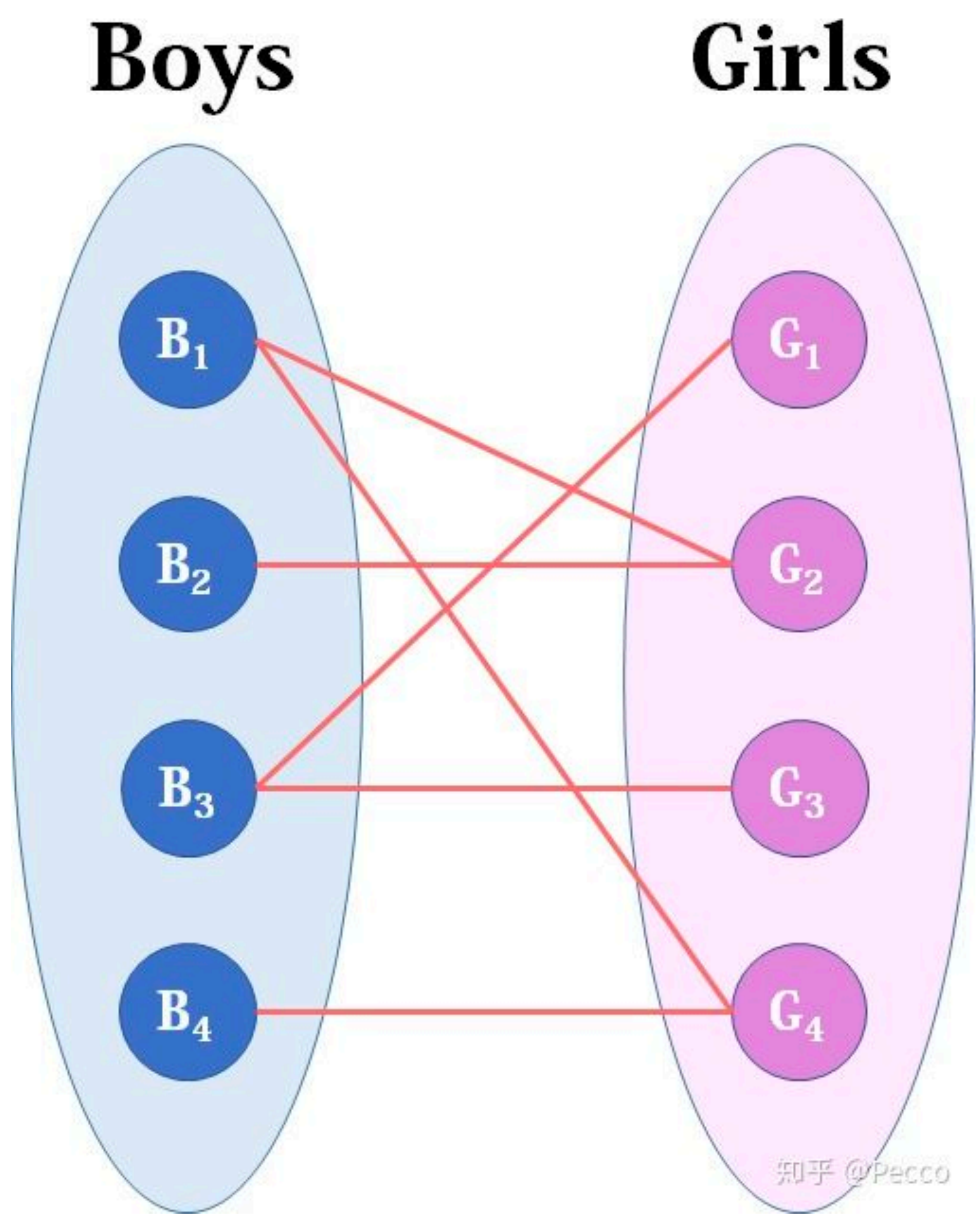
匈牙利算法

把二分图分为 A, B 两个集合，依次枚举 A 中的每个点，试图在B集合中找到一个匹配。

对于 A 集合中一点 x，假设 B 集合中有一个与其相连的点 y，若y暂时还没有匹配点，那么 x 可以和 y 匹配，找到；

否则，设 y 已经匹配的点为 z（显然 z 是 A 集合中的一个点），那么，我们将尝试为 z 找到一个除了 y 之外的匹配点，若找到，那么 x 可以和 y 匹配，否则 x 不能与 y 匹配。

二分图最大匹配



二分图最大匹配

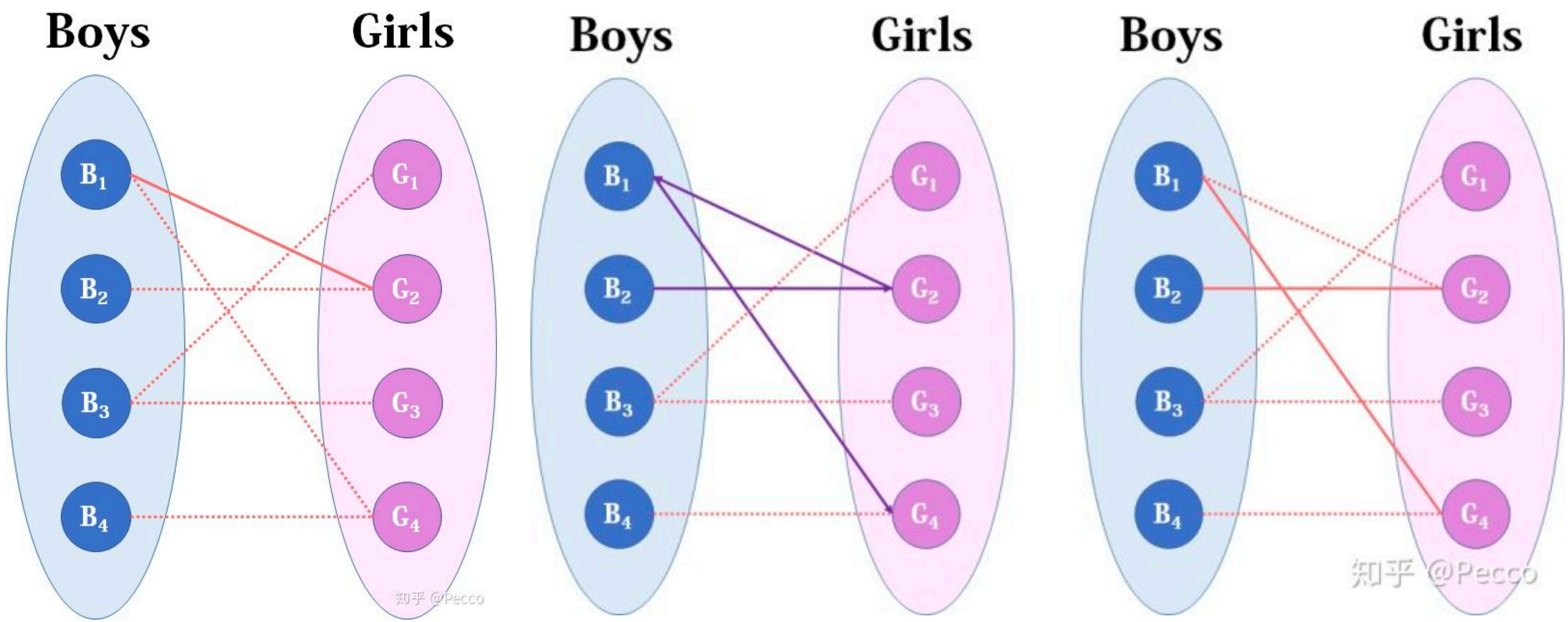


图 1：先把 B1 和 G2 连接；图 2：B2 尝试与 G2 连接，但是 G2 已经和 B1 连在一起，这时候我们倒回去看看 B1 还有没有其他的选择；图 3：发现可以给 B1 安排 G4，B2 安排 G2

二分图最大匹配

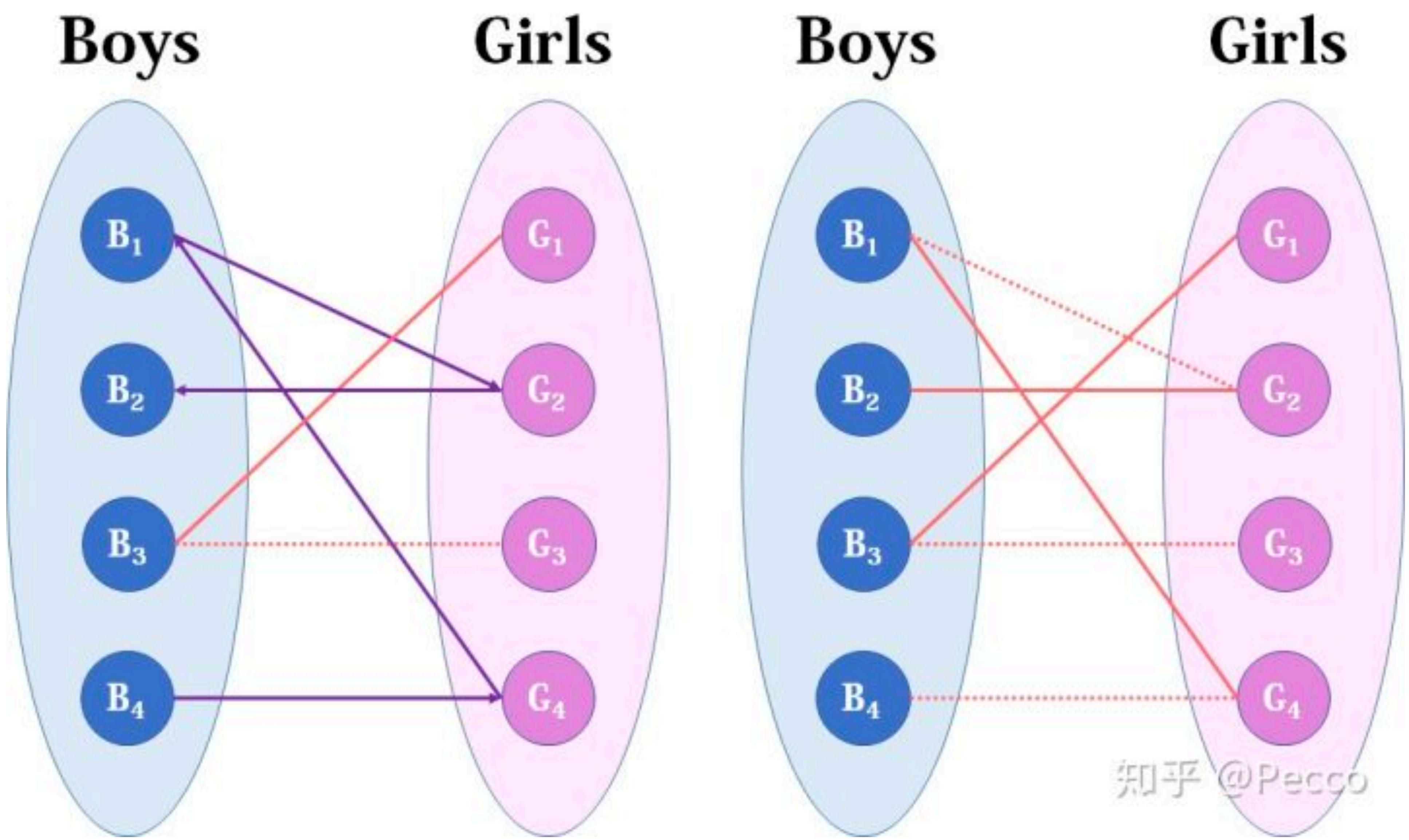


图 1：给 B3 安排上 G1；图 2: B4 只能选 G4，但是给 B1 重新分配一个已经不可能了

这就是匈牙利算法的过程，可以尝试一下 POJ1274

匈牙利算法

```
bool dfs(int x)
{
    for(int y = 1; y <= m; y++)
    {
        if(mp[x][y] && !vis[y])
        {
            vis[y] = 1;
            if(!lk[y] || dfs(lk[y]))
            {
                lk[y] = x;
                return 1;
            }
        }
    }
    return 0;
}
int main()
{
    while(scanf("%d%d", &n, &m) != EOF)
    {
        memset(mp, 0, sizeof(mp));
        for(int i = 1; i <= n; i++)
        {
            int t, x;
            scanf("%d", &t);
            while(t--)
            {
                scanf("%d", &x);
                mp[i][x] = 1;
            }
        }
        memset(lk, 0, sizeof(lk));
        int ans = 0;
        for(int i = 1; i <= n; i++)
        {
            memset(vis, 0, sizeof(vis));
            if(dfs(i))
                ans++;
        }
        cout << ans << endl;
    }
    return 0;
```

顶点覆盖和独立集

定义：假如选了一个点就相当于覆盖了以它为端点的所有边。最小顶点覆盖就是选择最少的点来覆盖所有的边。

定理：最小顶点覆盖等于二分图的最大匹配。

设最大匹配是M，考虑最小顶点覆盖：

（1）M个点是足够的。就是说他们覆盖最大匹配的那M条边后，假设有某边 e 没被覆盖，那么把 e 加入后会得到一个更大的匹配，出现矛盾。

（2）M个点是必需的。匹配的M条边，由于他们两两无公共点，就是说至少有M个点才能把他们覆盖。

顶点覆盖和独立集

独立集是一个点集，点集中的各点之间没有连边。

最大独立集 = 点的总数 - 最小顶点覆盖

如果去掉这些点， 相应的关系（边）也都没有了。剩下的点之间就相互没有关系，就变成了独立集。

练习题

CF-gym-101755D. Transfer Window

n 个球员，有 m 个关系，表示某个球员可以换成另外一个球员。现在有某 K 个球员，想要 K 个其它球员（可能已经有了可能没有），问是否有交换方案。

n <= 300, m <= 90000

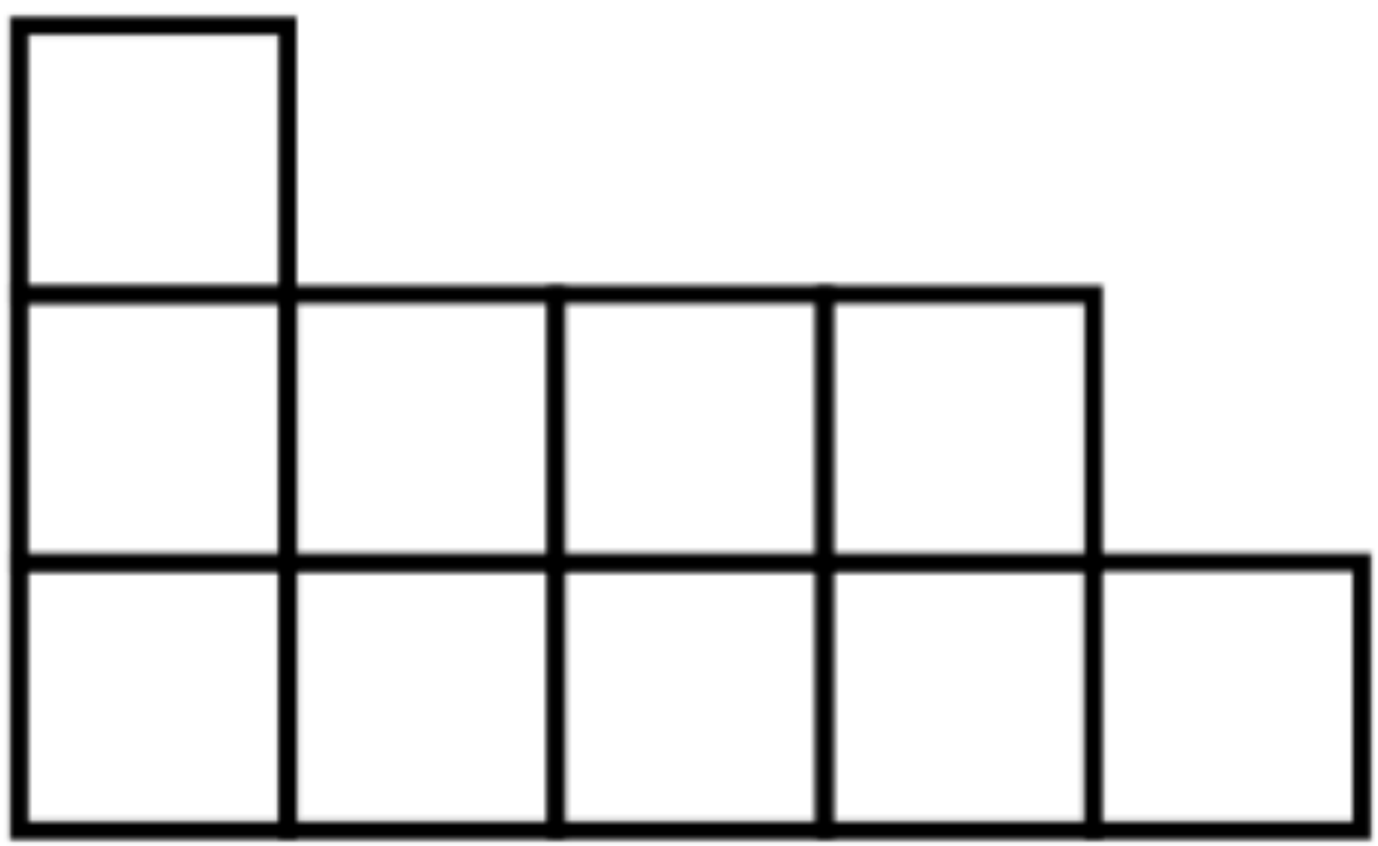
CF-gym-101755D. Transfer Window

Floyd 预处理闭包，建立二分图，左部是不想要的球员，右部是还没获得的球员，看是否能够满匹配。

CF1268B. Domino for Young

有 n 列格子，排在一起，第 i 列的高度是 ai，要在这个图上放 1x2 的多米诺骨牌，问最多可以放多少张骨牌?

n <= 300000, ai <= 300000 且单调递减



Young diagram for $a = [3, 2, 2, 2, 1]$.

CF1268B. Domino for Young

对格子进行黑白染色，答案是两种格子数量的较小值

考虑建二分图，容易证明匹配一定可以配满，因为任意一个白格到任意一个黑格之间都能找一个增广路

CF623A.Graph and String

n 个结点的无向图，每个结点标号”abc” 三个字母其中一个。

将标号为相同字母的结点连边，将所有标号为 b 的结点与其它标号的结点连边。

给出图的 m 个连边，求一种合法的标号方案，不存在输出’NO’。

$1 \leq n \leq 500, 1 \leq m \leq 200000$

CF623A.Graph and String

如果有一个点和其它点都有连边，将其标号 b。 然后图剩下两个团，一个标号 a，一个标号 c。

CF1093D. Beautiful Graph

n 个点 m 条边的无向图，可以给每个点赋权值 1, 2 或 3。

要求赋值之后，每条边的两个端点权值和是奇数，问有多少种赋值可能，答案对 1000000007 取模。

$1 \le n, m \le 100000$

CF1093D. Beautiful Graph

不同的连通块可以乘法原理算答案。

单点有 3 种可能赋值。

一个连通块是二分图是才有解，如果左部有 p 个点，右部有 q 个点，这个连通块有 2^p + 2^q 种方案数。

CF741C. Arpa's overnight party and Mehrdad's silent entering

有 n 对情侣坐成一个圈，有两种食物，要给每个人分其中一种，要求每对情侣的食物不同，任意连续的三个人必须要有两人食物不同。

求分配方案，无解输出-1

$1 \leq n \leq 100000$

CF741C. Arpa’s overnight party and Mehrdad’s silent entering

改变限制，要求 2i 和 2i – 1 食物类型不同

发现这张图上没有奇环

二分图染色