# 最大团和最大独立集

1.独立集：任意两点都不相连的顶点的集合

2.定理：最大独立集=顶点数-最大匹配边数（二分图的顶点数应该是左右两顶点之和）

3.完全子图：任意两点都相连的顶点的集合（最大完全子图即最大团）

4补图：原来图中有的边删除，没有的边加上(即关系矩阵完全取反)

5.定理：最大团=原图补图的最大独立集

6定理：最大独立集=原图补图的最大团

根据定理2，5我们可以得到，如果要求最大团且原图的补图为二分图，那么我可以用匈牙利求出补图的最大匹配数继而得到最大独立集。否则我们就只能暴搜，但我们可以使用A\*来优化（见1419）

Poj 3692

题意：在幼儿园中，有许多小孩。其中有男孩，也有女孩。女孩之间相互认识，男孩之间也相互认识。同时，一些男孩和女孩之间也相互认识，有一天，老师希望从所有人之中选出一些人来玩游戏，这个游戏需要所有的参与者之间相互认识，问老师可以最多找出多少人来玩这个游戏。

如果将男孩女孩看做顶点，男女之间的认识关系看做边，那么本题就转化为在图中求一个最大完全子图（最大团）的规模。但是在本题有特殊条件，男孩之间相互认识，女孩之间相互认识，即男孩和女孩本身就是两个完全子图，同时，如果将男孩和女孩看做集合，不能直接选取认识关系做边，因为这样违反了二分图的定义。那么我们可以采用**认识关系的补集**作为边，X= {男孩集合}， Y= {女孩集合}， E= {（i,j）| 男孩i与女孩j之间相互**不认识**}， 构建二分图G= {X,Y,E}。

最大独立集：从二分图G= （X,Y,E）中选取一些点V\*（属于X+Y），使得点集V\*中任意两点之间没有边相连。

最大独立点集元素个数 = 节点数（X+Y） - 最大匹配数；

由于E是认识关系的补集构成，那么该二分图的最大独立集元素的个数等价于原图中最大团的顶点数。

因此我们只需构造这样的二分图，利用匈牙利算法求出最大匹配数，得到最大独立集顶点个数就是答案。

[**POJ 1466**](http://blog.csdn.net/sdj222555/article/details/7426592)

有n个人，每个人与其他的某几个人有关系，这个关系且称为浪漫关系，然后最后求一个最大的集合，使得集合中所有的人两两之间都不存在浪漫关系。

看到之后就可以发现，这是一道非常明显的最大独立集的问题，可以转化为二分图来做，还是最经典的拆点建图，然后根据定理，最大独立集=顶点数-最小点覆盖数。  而对于这道题来说，我们可以发现这个浪漫关系是相互的。

而我们的建图中，按理来说应该是一边是男的点，一边是女的点这样连边，但是题目中没说性别的问题。

只能将每个点拆成两个点，**一个当作是男的点，一个当作是女的点了（无语..）**，然后连边。由于关系是相互的，这样就造成了边的重复。也就是边集是刚才的二倍，从而导致了最大匹配变成了二倍。

那么 ，最大独立集=顶点数-最大匹配/2，所以最终答案就呼之欲出了。

Poj1419

求裸的最大团

求最大独立集是一个NP问题，目前没有多项式时间内的解法，所以常见解法就是搜索。

我们可以使用A\*来优化

在main函数枚举中

for(int u=V;u>0;u--){//不断增大搜索的范围，用于得到估值函数

dfs(u,1);

the\_max[u]=ans;//得到估值函数中的一个值

}

在dfs中

for(int v=u+1;v<=V;++v)//剪枝：只选择比当前节点编号大的节点，避免重复搜索

if(the\_max[v]+num<=ans) continue;//the\_max[v]保存的是由编号只由比v大的点组成的子集的当前最优解，故最终答案是the\_max[1]

从而剪掉大量没用的搜索