最大团

```
#include<bits/stdc++.h>
using namespace std;

const int N = 105;

int n, G[N][N];
int cntClique, pts[N], res[N], cnt[N];

bool dfs(int pos, int num){
    for(int i=pos+1; i<=n; ++i){
        if(cnt[i]+num<=cntClique) return false;

        if(G[pos][i]){
            int ok=1;
            for(int id=1; id<=num; ++id){
                if(!G[i][pts[id]]){
                    ok=0;
                    break;
                }
            }

            if(ok){
                pts[num+1]=i;
                if(dfs(i,num+1)) return true;
            }
        }
    }

    if(num>cntClique){
        for(int i=1; i<=num; ++i){
            res[i]=pts[i];
        }
        cntClique=num;
        return true;
    }

    return false;
}

void maxClique(){
    cntClique=-1;
    for(int i=n; i>0; --i){
        pts[1]=i;
        dfs(i, 1);
        cnt[i]=cntClique;
    }
}

int main(){
    while(scanf("%d",&n) && n){
        memset(G, 0, sizeof G);
        for(int i=1; i<=n; ++i){
```

```
            for(int j=1; j<=n; ++j){
                scanf("%d", &G[i][j]);
            }
        }

        maxClique();
        printf("%d\n", cntClique);
    }
    return 0;
}
```

一般图最大匹配

```
// Problem: P6113 【模板】一般图最大匹配
// Contest: Luogu
// URL: https://www.luogu.com.cn/problem/P6113
// Memory Limit: 128 MB
// Time Limit: 1000 ms
//
// Powered by CP Editor (https://cpeditor.org)

#include<bits/stdc++.h>
using namespace std;

const int MN = 1010;

vector<int> g[MN];
int pa[MN],match[MN],root[MN],vis[MN],stp[MN];
int n,t;

int lca(int u,int v){
    for(t++;;swap(u,v)){
        if(u==0)    continue;
        if(stp[u]==t)   return u;
        stp[u]=t;
        u=root[pa[match[u]]];
    }
}

#define qpush(u) q.push(u),vis[u]=0
void flower(int u,int v,int l,queue<int>& q){
    while(root[u]!=l){
        pa[u]=v;
        if(vis[v=match[u]]==1)  qpush(v);
        root[u]=root[v]=l;
        u=pa[v];
    }
}

bool bfs(int u){
    for(int i=1;i<=n;i++)   root[i]=i;
    memset(vis,-1,sizeof vis);
    queue<int> q;
```

```
    qpush(u);
    while(!q.empty()){
        u=q.front();q.pop();
        for(int v:g[u]){
            if(vis[v]==-1){
                pa[v]=u;vis[v]=1;
                if(!match[v]){
                    for(int tmp;u;v=tmp,u=pa[v]){
                        tmp=match[u];
                        match[match[u]=v]=u;
                    }
                    return true;
                }
                qpush(match[v]);
            }
            else if(vis[v]==0&&root[u]!=root[v]){
                int l=lca(root[v],root[u]);
                flower(v,u,l,q);
                flower(u,v,l,q);
            }
        }
    }
    return 0;
}

int blossom(){
    memset(pa,0,sizeof pa);
    memset(match,0,sizeof match);
    int ans=0;
    for(int i=1;i<=n;i++)
        if(!match[i]&&bfs(i))
            ans++;
    return ans;
}

int main(){
    int m;
    cin>>n>>m;
    for(int i=1,x,y;i<=m;i++){
        cin>>x>>y;
        g[x].emplace_back(y);
        g[y].emplace_back(x);
    }
    cout<<blossom()<<endl;
    for(int i=1;i<=n;i++)
        cout<<match[i]<<' ';
    cout<<endl;
}
```