

题意

主持人总共有 m 道题，选手有 n 种不同的“锦囊妙计”。主持人规定，每道题都可以从两种“锦囊妙计”中选择一种，而每种“锦囊妙计”只能用一次。我们又假设一道题使用了它允许的锦囊妙计后，就一定能正确回答，顺利进入下一题。现在我来到了节目现场，可是我实在是太笨了，以至于一道题也不会做，每道题只好借助使用“锦囊妙计”来通过。如果我事先就知道了每道题能够使用哪两种“锦囊妙计”，那么你能告诉我怎样选择才能通过最多的题数吗？

分析

左边的点为题目 右边的点为锦囊，对于每一个题目，向对应的锦囊连边，最后求左右的最大匹配

代码

```
1  #include<bits/stdc++.h>
2  using namespace std;
3  typedef long long LL;
4  vector<int>g[1001];
5  int n,m;
6  bool use[1001];
7  int from[1001];
8  bool match(int x)
9  {
10     for(int i=0;i< g[x].size();++i)
11     {
12         int to = g[x][i];
13         if(!use[to])
14         {
15             use[to] = true;
16             if(!from[to] || match(from[to]))
17             {
18                 from[to] = x;
19                 return true;
20             }
21         }
```

```
22     }
23     return false;
24 }
25 int main()
26 {
27     ios::sync_with_stdio(false);
28     cin>>n>>m;
29     int x,y;
30     for(int i=1;i<=m;i++)
31     {
32         cin>>x>>y;
33         g[i].push_back(x);
34         g[i].push_back(y);
35     }
36     int ans = 0;
37     for(int i=1;i<=m;i++)
38     {
39         memset(use,0,sizeof(use));
40         if(match(i)) ans++;
41         else break;
42     }
43     cout<<ans<<endl;
44     return 0;
45 }
```