

224프 (
소스코드

최백준 choi@startlink.io



C++14

```
1 #include <iostream>
2 #include <vector>
3 #include <algorithm>
4 using namespace std;
5 bool a[2000][2000];
6 vector<int> g[2000];
7 vector<pair<int,int>> edges;
8 int main() {
9     int n, m;
10    cin >> n >> m;
11    for (int i=0; i<m; i++) {
12        int from, to;
13        cin >> from >> to;
14        edges.push_back({from, to});
15        edges.push_back({to, from});
16        a[from][to] = a[to][from] = true;
17        g[from].push_back(to);
18        g[to].push_back(from);
19    }
20    m *= 2;
21    for (int i=0; i<m; i++) {
22        for (int j=0; j<m; j++) {
23            // A -> B
24            int A = edges[i].first;
25            int B = edges[i].second;
26            // C -> D
27            int C = edges[j].first;
28            int D = edges[j].second;
29            if (A == B || A == C || A == D || B == C || B == D || C == D) {
30                continue;
31            }
32            // B -> C
33            if (!a[B][C]) {
34                continue;
35            }
36            // D -> E
37            for (int E : g[D]) {
38                if (A == E || B == E || C == E || D == E) {
39                    continue;
40                }
41                cout << 1 << '\n';
42                return 0;
43            }
44        }
45    }
46    cout << 0 << '\n';
47    return 0;
48 }
```

인접행렬
인접리스트
간선 리스트

ABC②-E

결과

메모리

시간

코드 길이

맞았습니다!!

5940 KB

88 ms

1223 B

C++14

```
1 #include <cstdio>
2 #include <vector>
3 using namespace std;
4 vector<int> a[1001];
5 bool check[1001];
6 void dfs(int node) {
7     check[node] = true;
8     for (int i=0; i<a[node].size(); i++) {
9         int next = a[node][i];
10        if (check[next] == false) {
11            dfs(next);
12        }
13    }
14 }
15 int main() {
16     int n, m;
17     scanf("%d %d",&n,&m);
18     for (int i=0; i<m; i++) {
19         int u,v;
20         scanf("%d %d",&u,&v);
21         a[u].push_back(v);
22         a[v].push_back(u);
23     }
24     int components = 0;
25     for (int i=1; i<=n; i++) {
26         if (check[i] == false) {
27             dfs(i);
28             components += 1;
29         }
30     }
31     printf("%d\n",components);
32     return 0;
33 }
```

결과

메모리

시간

코드 길이

맞았습니다!!

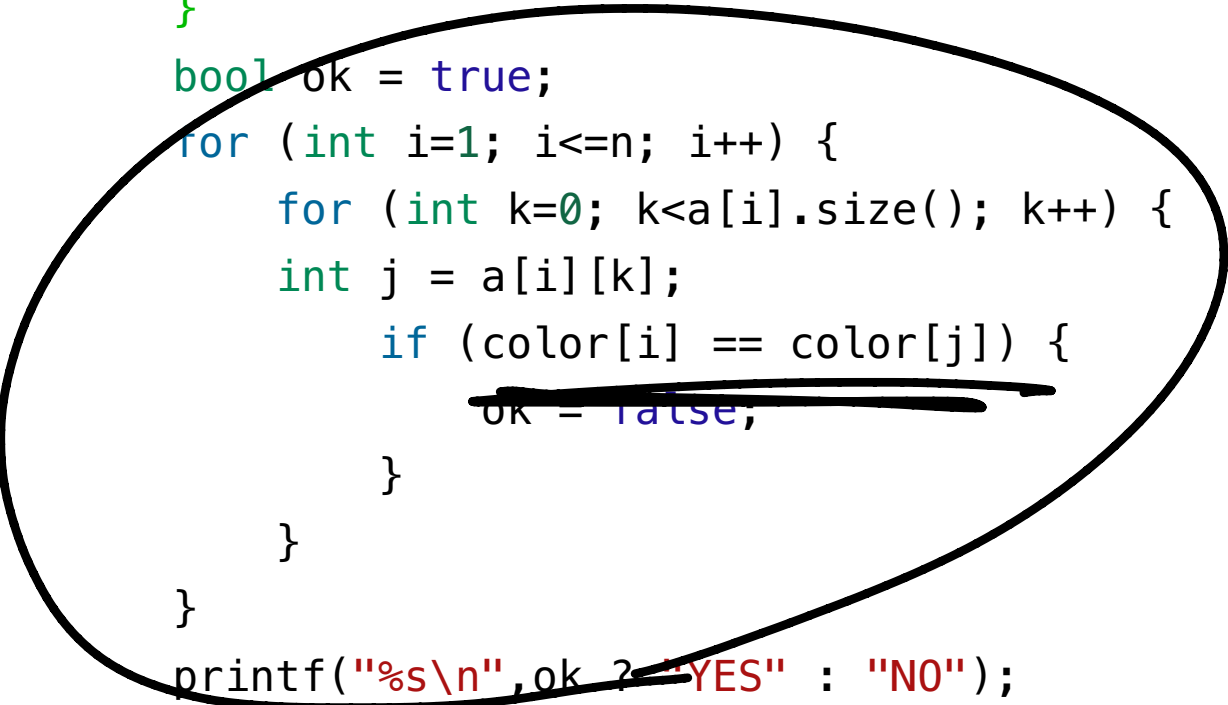
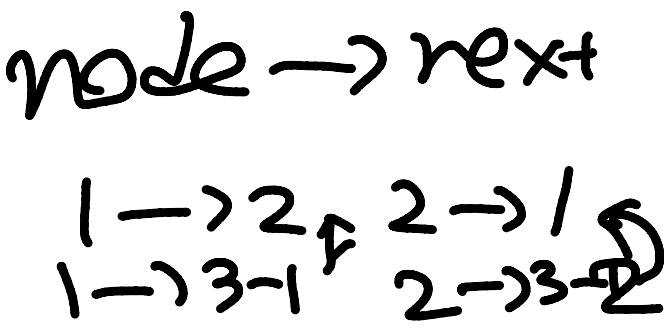
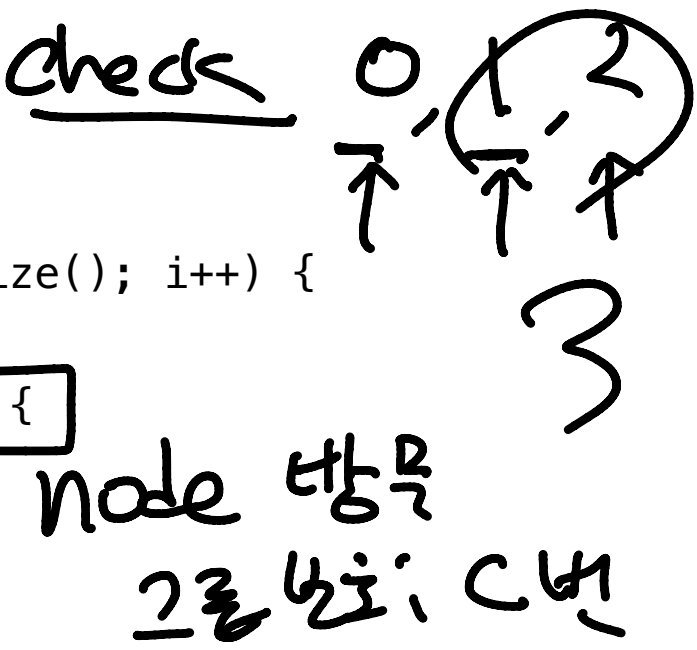
5556 KB

132 ms

681 B

C++14

```
1 #include <cstdio>
2 #include <algorithm>
3 #include <cstring>
4 #include <vector>
5 #include <queue>
6 using namespace std;
7 vector<int> a[20001];
8 int color[20001];
9 void dfs(int node, int c) {
10     color[node] = c;
11     for (int i=0; i<a[node].size(); i++) {
12         int next = a[node][i];
13         if (color[next] == 0) {
14             dfs(next, 3-c);
15         }
16     }
17 }
18 int main() {
19     int t;
20     scanf("%d\n",&t);
21     while (t--) {
22         int n, m;
23         scanf("%d %d",&n,&m);
24         for (int i=1; i<=n; i++) {
25             a[i].clear();
26             color[i] = 0;
27         }
28         for (int i=0; i<m; i++) {
29             int u,v;
30             scanf("%d %d",&u,&v);
31             a[u].push_back(v);
32             a[v].push_back(u);
33         }
34         for (int i=1; i<=n; i++) {
35             if (color[i] == 0) {
36                 dfs(i, 1);
37             }
38         }
39         bool ok = true;
40         for (int i=1; i<=n; i++) {
41             for (int k=0; k<a[i].size(); k++) {
42                 int j = a[i][k];
43                 if (color[i] == color[j]) {
44                     ok = false;
45                 }
46             }
47         }
48         printf("%s\n",ok ? "YES" : "NO");
49     }
50     return 0;
51 }
```



결과	메모리	시간	코드 길이
맞았습니다!!	10432 KB	324 ms	1161 B

C++14

```
1 #include <cstdio>
2 #include <algorithm>
3 #include <cstring>
4 #include <vector>
5 #include <queue>
6 using namespace std;
7 vector<int> a[20001];
8 int color[20001];
9 bool dfs(int node, int c) {
10     color[node] = c;
11     for (int i=0; i<a[node].size(); i++) {
12         int next = a[node][i];
13         if (color[next] == 0) {
14             if (dfs(next, 3-c) == false) {
15                 return false;
16             }
17         } else if (color[next] == color[node]) {
18             return false;
19         }
20     }
21     return true;
22 }
23 int main() {
24     int t;
25     scanf("%d\n",&t);
26     while (t--) {
27         int n, m;
28         scanf("%d %d",&n,&m);
29         for (int i=1; i<=n; i++) {
30             a[i].clear();
31             color[i] = 0;
32         }
33         for (int i=0; i<m; i++) {
34             int u,v;
35             scanf("%d %d",&u,&v);
36             a[u].push_back(v);
37             a[v].push_back(u);
38         }
39         bool ok = true;
40         for (int i=1; i<=n; i++) {
41             if (color[i] == 0) {
42                 if (dfs(i, 1) == false) {
43                     ok = false;
44                 }
45             }
46         }
47         printf("%s\n",ok ? "YES" : "NO");
48     }
49     return 0;
50 }
```

014 t/f

node → next

↑

결과

메모리

시간

코드 길이

맞았습니다!!

10432 KB

320 ms

1147 B

끝

코드 플러스

<https://code.plus>

- 슬라이드에 포함된 소스 코드를 보려면 "정보 수정 > 백준 온라인 저지 연동"을 통해 연동한 다음, "백준 온라인 저지"에 로그인해야 합니다.
- 강의 내용에 대한 질문은 코드 플러스의 "질문 게시판"에서 할 수 있습니다.
- 문제와 소스 코드는 슬라이드에 첨부된 링크를 통해서 볼 수 있으며, "백준 온라인 저지"에서 서비스됩니다.
- 슬라이드와 동영상 강의는 코드 플러스 사이트를 통해서만 볼 수 있으며, 동영상 강의의 녹화와 다운로드, 배포와 유통은 저작권법에 의해서 금지되어 있습니다.
- 다른 경로로 이 슬라이드나 동영상 강의를 본 경우에는 codeplus@startlink.io 로 이메일 보내주세요.
- 강의 내용, 동영상 강의, 슬라이드, 첨부되어 있는 소스 코드의 저작권은 스타트링크와 최백준에게 있습니다.