

Contest Duration: 2025-06-28(Sat) 08:00 (<http://www.timeanddate.com/worldclock/fixedtime.html?iso=20250628T2100&p1=248>) - 2025-06-28(Sat) 09:40 (<http://www.timeanddate.com/worldclock/fixedtime.html?iso=20250628T2240&p1=248>) (local time) (100 minutes)

[Back to Home \(/home\)](/home)
[🏠 Top \(/contests/abc412\)](/contests/abc412)
[☰ Tasks \(/contests/abc412/tasks\)](/contests/abc412/tasks)
[❓ Clarifications \(/contests/abc412/clarifications\)](/contests/abc412/clarifications)
[🚀 Submit \(/contests/abc412/submit\)](/contests/abc412/submit)
[☰ Results ▾](/contests/abc412/standings)
[⬇️ Standings \(/contests/abc412/standings\)](/contests/abc412/standings)
[⬇️ Virtual Standings \(/contests/abc412/standings/virtual\)](/contests/abc412/standings/virtual)
[🔧 Custom Test \(/contests/abc412/custom\\_test\)](/contests/abc412/custom_test)
[📖 Editorial \(/contests/abc412/editorial\)](/contests/abc412/editorial)
[💬 Discuss \(https://codeforces.com/blog/entry/144250\)](https://codeforces.com/blog/entry/144250)


Official

## D - Make 2-Regular Graph

### (/contests/abc412/tasks/abc412\_d) Editorial by en\_translator (/users/en\_translator)

For a simple undirected graph, the degree of every vertex is 2 if and only if every connected component is a cycle graph. Since the graph is simple, each cycle graph must have three or more vertices; thus, any connected component has at least three vertices.

By the constraints  $N \leq 8$ , there are either one or two connected components in the final graph. Also note that, given a final graph, the minimum number of operations is easily counted by enumerating the edges that exists exactly one of either the initial or the final graph. We discuss by cases: when there are one connected component, and when there are two.

#### (1) if there is one connected component

It is sufficient to consider all permutations  $(P_1, P_2, \dots, P_N)$  of  $(1, 2, \dots, N)$  and consider the cycle graph of this order (where there are edges between  $P_i$  and  $P_{i+1}$ , with  $P_{N+1} := P_1$ ). We may fix  $P_1 = 1$ , but the small constraints allow us to enumerate all permutations.

#### (1) if there are two connected components

It is fast enough to try all divisions into two connected components and do the same thing as (1). On implementation, we may perform an exhaustive search using bitmasks, or enumerate permutations just as (1), split into the first  $d$  elements and last  $(N - d)$  elements, and regard each portion as a cycle.

Sample code

2025-06-30 (Mon)

00:07:51 -04:00

```

1. #include <bits/stdc++.h>
2. using namespace std;
3. #define rep(i, n) for (int i = 0; i < (n); i++)
4.
5. int main() {
6.     int n, m;
7.     cin >> n >> m;
8.     vector f(n, vector<bool>(n));
9.     rep(i, m) {
10.         int u, v;
11.         cin >> u >> v;
12.         u--; v--;
13.         if (u > v) swap(u, v);
14.         f[u][v] = true;
15.     }
16.     int ans = 100;
17.     vector<int> a(n);
18.     rep(i, n) a[i] = i;
19.     do {
20.         // cycle * 1
21.         vector g(n, vector<bool>(n));
22.         rep(i, n) {
23.             int u = a[i], v = a[(i + 1) % n];
24.             if (u > v) swap(u, v);
25.             g[u][v] = true;
26.         }
27.         int c0 = 0;
28.         rep(i, n) rep(j, n) if (f[i][j] != g[i][j]) c0++;
29.         if (c0 < ans) ans = c0;
30.
31.         // cycle * 2
32.         for (int d = 3; d <= n - 3; d++) {
33.             vector h(n, vector<bool>(n));
34.             rep(i, d) {
35.                 int u = a[i], v = a[(i + 1) % d];
36.                 if (u > v) swap(u, v);
37.                 h[u][v] = true;
38.             }
39.             rep(i, n - d) {
40.                 int u = a[i + d], v = a[(i + 1) % (n - d) + d];
41.                 if (u > v) swap(u, v);
42.                 h[u][v] = true;
43.             }
44.             int c1 = 0;
45.             rep(i, n) rep(j, n) if (f[i][j] != h[i][j]) c1++;
46.             if (c1 < ans) ans = c1;
47.         }

```

2025-06-30 (Mon)  
00:07:51 -04:00

```
48.         } while (next_permutation(a.begin(), a.end()));
49.
50.         cout << ans << '\n';
51. }
```

posted: a day ago

last update: a day ago

---

am)

ps%3A%2F%2Fatcoder.jp%2Fcontests%2Fabc412%2Feditorial%2F13394%3Flang%3Den&title=Editorial%20-3412)

---

[Rule \(/contests/abc412/rules\)](/contests/abc412/rules) [Glossary \(/contests/abc412/glossary\)](/contests/abc412/glossary)

[Terms of service \(/tos\)](/tos) [Privacy Policy \(/privacy\)](/privacy) [Information Protection Policy \(/personal\)](/personal) [Company \(/company\)](/company)

[FAQ \(/faq\)](/faq) [Contact \(/contact\)](/contact)

Copyright Since 2012 ©AtCoder Inc. (<http://atcoder.co.jp>) All rights reserved.

2025-06-30 (Mon)  
00:07:51 -04:00