

1. Tìm đường trong mê cung

Nhiệm vụ: Cho đồ thị vô hướng và hai đỉnh u và v , kiểm tra xem liệu có đường đi giữa u và v .

Định dạng đầu vào: Đồ thị với n đỉnh và m cạnh. Dòng tiếp theo chứa hai đỉnh u và v của đồ thị.

Ràng buộc: $2 \leq n \leq 10^3$; $1 \leq m \leq 10^3$; $1 \leq u, v \leq n$; $u \neq v$.

Định dạng đầu ra: Đưa ra 1 nếu có đường đi giữa u và v ; và 0 nếu không có.

Mã nguồn mẫu. Bạn hãy viết thêm mã vào hàm reach.

```
1 #include <iostream>
2 #include <vector>
3 using namespace std;
4
5 int reach(vector<vector<int>> &adj, int x, int y) {
6     //write your code here
7     return 0;
8 }
9
10 int main() {
11     int n, m;
12     cin >> n >> m;
13     vector<vector<int>> adj(n, vector<int>());
14     for (int i = 0; i < m; i++) {
15         int x, y;
16         cin >> x >> y;
17         adj[x - 1].push_back(y - 1);
18         adj[y - 1].push_back(x - 1);
19     }
20     int x, y;
21     cin >> x >> y;
22     cout << reach(adj, x - 1, y - 1);
23 }
```

Ví dụ 1: Dữ liệu vào:

```
4 4
1 2
3 2
4 3
1 4
1 4
```

Dữ liệu ra:

1

Ví dụ 2: Dữ liệu vào:

```
4 2
1 2
3 2
1 4
```

Dữ liệu ra:

0

Vào ra dữ liệu: Dữ liệu được nhập stdin và xuất ra stdout.

2. Kiểm tra chu trình

Nhiệm vụ: Kiểm tra xem liệu một **đồ thị có hướng** với n đỉnh và m cạnh có chứa chu trình.

Định dạng dữ liệu vào: Đồ thị có hướng được cho ở dạng danh sách cạnh.

Ràng buộc: $2 \leq n \leq 10^3$; $1 \leq m \leq 10^3$; $1 \leq u, v \leq n$; $u \neq v$.

Định dạng đầu ra: Đưa ra 1 nếu đồ thị có chu trình; và 0 nếu không có.

2.1. Mã nguồn mẫu. Bạn hãy viết thêm mã nguồn vào hàm `acyclic`.

```
1 #include <iostream>
2 #include <vector>
3 using namespace std;
4
5 int acyclic(vector<vector<int>> &adj) {
6     //write your code here
7     return 0;
8 }
9
10 int main() {
11     int n, m;
12     cin >> n >> m;
13     vector<vector<int>> > adj(n, vector<int>());
14     for (int i = 0; i < m; i++) {
15         int x, y;
16         cin >> x >> y;
17         adj[x - 1].push_back(y - 1);
18     }
19     cout << acyclic(adj);
20 }
```

Ví dụ 1: Dữ liệu vào:

```
4 4
1 2
4 1
2 3
3 1
```

Dữ liệu ra:

1

Ví dụ 2: Dữ liệu vào:

```
5 7
1 2
2 3
1 3
3 4
1 4
2 5
3 5
```

Dữ liệu ra:

0

Vào ra dữ liệu: Dữ liệu được nhập stdin và xuất ra stdout.

3. Khoảng cách giữa hai đỉnh trong đồ thị vô hướng

Nhiệm vụ: Cho một đồ thị vô hướng với n đỉnh và m cạnh và hai đỉnh u và v , hãy tính độ dài của đường đi ngắn nhất giữa u và v .

Định dạng dữ liệu vào: Đồ thị có hướng được cho ở dạng danh sách cạnh. Dòng tiếp theo chứa hai đỉnh u và v .

Ràng buộc: $2 \leq n \leq 10^5$; $1 \leq m \leq 10^5$; $1 \leq u, v \leq n$; $u \neq v$.

Định dạng đầu ra: Đưa ra số cạnh của đường đi ngắn nhất từ u tới v , hoặc -1 nếu không có đường đi.

Ví dụ 1: Dữ liệu vào:

```
4 4
1 2
4 1
2 3
3 1
2 4
```

Dữ liệu ra:

```
2
```

Ví dụ 2: Dữ liệu vào:

```
5 4
5 2
1 3
3 4
1 4
3 5
```

Dữ liệu ra:

```
-1
```

Vào ra dữ liệu: Dữ liệu được nhập stdin và xuất ra stdout.