

**Lab VII**  
11. Mar. 2016

---

**Problem 1**

Given an undirected graph, you are required to print the vertices of the graph using Breadth First Traversal.

**Input**

Your program will be tested against multiple test cases. Each test case begins with two integers  $n$  and  $e$ , representing the number of vertices and edges. The next  $e$  lines represent the vertices that are connected by an edge.

**Output**

For each test case, print the graph using BFS.

**Sample Input**

**6 6**  
0 1  
0 3  
1 2  
2 4  
3 4  
3 5

**5 7**  
0 1  
0 3  
0 4  
1 2  
1 3  
2 3  
3 4

**Sample Output**

0 1 3 2 4 5  
0 1 3 4 2

## **Problem 2**

Given an undirected graph, you are required to check if the graph is connected or not.

### **Input**

Your program will be tested against multiple test cases. Each test case begins with two integers  $n$  and  $e$ , representing the number of vertices and edges. The next  $e$  lines represent the vertices that are connected by an edge.

### **Output**

For each test case, print “connected” if the graph is connected; else print “not connected”.

#### **Sample Input**

```
6 6
0 1
0 3
1 2
2 4
3 4
3 5
```

```
7 5
0 1
0 2
1 2
3 4
5 6
```

#### **Sample Output**

```
connected
not connected
```

## **Problem 3**

Given an undirected graph, you are required to find the number of connected components.

### **Input**

Your program will be tested against multiple test cases. Each test case begins with two integers  $n$  and  $e$ , representing the number of vertices and edges. The next  $e$  lines represent the vertices that are connected by an edge.

### **Output**

For each test case, print the number of connected components.

**Sample Input**

6 4  
0 1  
0 2  
3 4  
4 5

8 4  
0 1  
2 3  
4 5  
6 7

**Sample Output**

2  
4

**Problem 4**

Given an undirected graph, you are required to check if the graph is bipartite.

**Input**

Your program will be tested against multiple test cases. Each test case begins with two integers  $n$  and  $e$ , representing the number of vertices and edges. The next  $e$  lines represent the vertices that are connected by an edge.

**Output**

For each test case, print “bipartite” if the graph is bipartite; else print “not bipartite”.

**Sample Input**

6 6  
0 1  
0 3  
1 2  
2 4  
3 4  
3 5

7 6  
0 1  
0 2  
0 3  
1 4  
2 5

**Sample Output**

not bipartite  
bipartite

**Problem 5**

Given an undirected graph, you are required to find the connected component with the largest number of vertices.

**Input**

Your program will be tested against multiple test cases. Each test case begins with two integers  $n$  and  $e$ , representing the number of vertices and edges. The next  $e$  lines represent the vertices that are connected by an edge.

**Output**

For each test case, print the number of vertices in the largest connected component.

**Sample Input**

**6 4**  
0 1  
0 2  
3 4  
4 5

**7 5**  
0 1  
0 2  
1 3  
2 3  
5 6

**Sample Output**

3  
4