LEBANESE AMERICAN UNIVERSITY
School of Arts and Science
Department of Computer Science and Mathematics

CSC 310: Algorithms and Data Structures

Lab VIII

Important Note: Input should be read from a file named "graph.in" for all the probles.

Problem 1

Given an undirected graph G (V, E), where V is the set of vertices and E is the set of edges, you are required to print the degree of each vertex.

Input

The first line is an integer *T* representing the number of test cases.
Each test case is made up of two integers n and *m* representing the number of vertices and edges respectively. Then *m* lines follow, each containing two integers *u* and *v* which are the endpoints of edge uv.

Output

For each test case, output the degree of each vertex in the graph (as in the below sample).

Sample Input	**Sample Output**
2	0: 2
5 5	1: 2
0 1	2: 2
0 2	3: 3
1 3	4: 1
2 3	
3 4	0: 2
5 5	1: 3
0 1	2: 3
0 2	3: 1
1 2	4: 1
1 3	
2 4	

Problem 2

Given an undirected graph G = (V, E), where V is the set of vertices and E is the set of edges, you are required to print the vertices of the graph using BFS starting with the vertex of highest index.

Input

The first line is an integer *T* representing the number of test cases.
Each test case is made up of two integers n and *m* representing the number of vertices and edges respectively. Then *m* lines follow, each containing two integers *u* and *v* which are the endpoints of edge uv.

Output

For each test case, list the vertices of the graph via a BFS traversal.

Sample Input

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

Sample Output

```
4 2 0 3 1
4 2 0 1 3
```

Problem 3

Given an undirected graph G = (V, E), you are required to print the vertices of G using DFS starting with the vertex of index zero.

Input

The first line is an integer *T* representing the number of test cases.

Each test case is made up of two integers n and *m* representing the number of vertices and edges respectively. Then *m* lines follow, each containing two integers *u* and *v* which are the endpoints of edge uv.

Output

For each test case, print the vertices according to a DFS traversal.

Sample Input	**Sample Output**
2 4 3 0 1 1 2 1 3 4 6 0 1 0 2 0 3 1 2 1 3 2 3	1 0 2 3 0 1 2 3

Problem 4

Given an undirected graph G = (V, E), print the connected components in the G.

Input

The first line is an integer *T* representing the number of test cases.
Each test case is made up of two integers n and *m* representing the number of vertices and edges respectively. Then *m* lines follow, each containing two integers *u* and *v* which are the endpoints of edge uv.

Output

For each test case, print each of the connected components on a separate line as shown in the below sample.

Sample Input	**Sample Output**
2	0: 0 1 2

5	4	1: 3 4
0	1	0: 0 1 2
0	2	1: 3 4
1	2	2: 5 6 7 8
3	4	
9	8	
0	1	
0	2	
1	2	
3	4	
5	6	
5	7	
6	8	
7	8	

Problem 5

Given a directed graph G (V, E), where V is the set of vertices and E is the set of edges, you are required to check if G is bipartite. A graph is bipartite if the graph can be colored using only 2 colors such that no adjacent vertices have the same color.

Input

The first line is an integer *T* representing the number of test cases.
Each test case is made up of two integers n and *m* representing the number of vertices and edges respectively. Then *m* lines follow, each containing two integers *u* and *v* which are the endpoints of edge uv.

Output

For each test case, output "**bipartite**" if the graph is bipartite and "**not bipartite**" otherwise.

Sample Input		**Sample Output**
2		bipartite
4	5	not bipartite
0	1	
1	2	
2	0	
2	3	
3	1	

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

Problem 6

Given an undirected graph G (V, E), where V is the set of vertices and E is the set of edges, you are required to check if the graph has a cycle.

Input

The first line is an integer *T* representing the number of test cases.
Each test case is made up of two integers n and *m* representing the number of vertices and edges respectively. Then *m* lines follow, each containing two integers *u* and *v* which are the endpoints of edge uv.

Output

For each test case, output "**cycle**" if the graph has a cycle and "**no cycle**" otherwise.

Sample Input	**Sample Output**
2	no cycle
5 3	cycle
0 4	
1 2	
2 3	
7 6	
0 1	
0 2	
0 3	
1 4	
2 5	
3 6	

Problem 7

Given an undirected graph G (V, E), where V is the set of vertices and E is the set of edges, you are required to check if the graph has a cycle of length greater than 4.

Input

The first line is an integer *T* representing the number of test cases.
Each test case is made up of two integers n and *m* representing the number of vertices and edges respectively. Then *m* lines follow, each containing two integers *u* and *v* which are the endpoints of edge uv.

Output

For each test case, output "yes" if the graph has a cycle of length > 4, and "no" otherwise.

Sample Input	**Sample Output**
2 5 3 0 4 1 2 2 3 5 7 0 1 0 2 0 3 0 4 1 3 3 4 4 2	no yes