# IT-301 (Design & Analysis of Algorithms ): Lab #8

1. Draw a simple connected directed graph with 8 vertices and 16 edges such that the in-degree and the out-degree of each vertex is 2. Show that there is a single non-simple cycle that includes all the edges of your graph.

2. Suppose we are given an undirected graph $G$ with $n$ vertices and $m$ edges. How many back edges will the DFS on $G$ produce.

3. Is it possible to draw a graph $G$ with 12 vertices, 66 edges, and 3 connected components. Give a brief explanation in support of your answer.

4. Explain why the DFS traversal runs in $\Theta(n^2)$ time on an $n$-vertex simple graph that is represented with an adjacency matrix.

5. Let $G$ be an undirected graph with $n$ vertices and $m$ edges. Prove that if $G$ is connected then $m \geq (n-1)$.

6. Prove that after performing DFS on an undirected connected graph, the discovery-edges form a spanning tree.

7. Given a directed graph $G = (V, E)$, how will you determine in $O(|V| + |E|)$ time whether the graph is strongly connected or not.

8. Give two different algorithms for finding the transitive closure of a graph, and compare their complexities.

9. Prove that a digraph admits a topological ordering iff it is a DAG.

10. Let $T$ be the spanning tree rooted at the start vertex produced by the DFS of a connected, undirected graph $G$. Argue why every edge of $G$, not in $T$, is a back-edge.