---

1. Prove or disprove: There are an even number of odd-degree vertices in any undirected graph.
   (Degree of a vertex is the number of edges incident on that vertex.)

2. In the lectures, we learnt that the two main graph representations are adjacency list and adjacency matrix. Design algorithms for the following tasks:

   (a) Given a graph $G = (V, E)$ as input in the adjacency matrix representation, output the adjacency list representation of G.

   (b) Given a graph $G = (V, E)$ as input in the adjacency list representation, output the adjacency matrix representation of G.

   Discuss running time of your algorithms.

3. The reverse of a directed graph $G = (V, E)$ is another directed graph $G^R = (V, E^R)$ on the same vertex set but with all the edges reversed. Design an algorithm that outputs the adjacency list of the reverse of a given graph G. G given as input in adjacency list format. Discuss running time.

4. An undirected graph is said to be *bipartite* iff its vertices can be partitioned into two sets such that there are no edges between any two vertices in the same partition.

   Design an algorithm to determine if a given undirected graph is bipartite. Give proof of correctness and running time analysis.