Advanced Data Networks
Assignment I
Due: 15 Esfand 1395

1. Given the incident matrix *A* of graph $G(N.L)$, find the adjacency matrix *g* of this graph (provide a mathematical relation not an algorithm) .

2. Consider graph G(N,L) with adjacency matrix *g*
   a. Using the adjacency matrix and its powers ($g.g^2.g^3.\ldots$), find whether a node is isolated in a graph or not.
   b. Come up with an algorithm which finds whether a graph is connected or not.
   c. Write an algorithm which finds all the components of a graph.
   d. Find the complexity of your algorithm in part c.

3. Using the Brute-force algorithm, find the complexity of obtaining clique number for a graph with $/N/$ nodes and $/L/$ links.

4. Consider a tandem wireless network with *6* nodes where the distance between any two neighbors is R. Also assume that the transmission range of all the nodes is R. Consider the interference protocol model. Plot the conflict graph. What is the minimum number of channels needed to schedule all possible links in the network? All the links are bi-directional.

5. Cliques
   a. Write down a greedy heuristic to find an approximate for the maximum clique.
   b. What is the complexity of your algorithm in part (a).
   c. Local search heuristic methods: this algorithm tries to improve your approximate maximal clique found in part (a). Let *C* denote the maximal clique found in part (a). In local search heuristic, we pick a member $c_1 \in C$ and then remove it from the maximal clique *C*. Since the obtaining set $(C \backslash c_i)$ is not a maximal clique any more, we try to make it a maximal clique by adding new nodes to this set if possible. We do that for the all members of *C* one by one. For every member of *C* that we remove, we obtain a new maximal clique if possible. Find the complexity of this algorithm.

Programming Assignment:

Write two MATLAB functions which accept the adjacency matrix of a network and

- Finds the maximum independent set using the brute-force algorithm.
- Find the maximum independent set using the Luby's algorithm while the minimum degree used for the algorithm is the minimum degree of the original grah.
- Find the maximum independent set using the Luby's algorithm while the minimum degree is updated in every iteration of the algorithm.

Test your functions using a random graph with 12 nodes and link probability 1/4.

Find the sub-optimality of the Luby's algorithm as follows: run brute-force function once and save the size of the independent set as s. Run the Luby's function 1000 times and finds the average size of the maximum independent set. Lets' call it q. The sub-optimality of the Luby's algorithm is q/s.