Design and Analysis of Algorithms 2013
(Home work 5)

November 15, 2013

- Due on, November 29, before 10 am.
- Late home works will not be accepted.
- Please give precise arguments for all statements that you write.
- Please do not hesitate to contact me if you do not understand the problems.
- Collaboration is discouraged, but not prohibited. It is recommended that you try to solve the problems on your own. You can discuss the questions with your colleagues but you should not copy solutions. Always write down your own answers. If copying is detected that may immediately lead to a grade less than 7. (**This would be followed strictly**)
- Credits would be given to partial solutions also.
- When you write an algorithm, you should briefly discuss the main idea of your algorithm, then write a pseudo code, argue about its correctness and state and prove the running time of your algorithm.
- The answers should be typed or written clearly and a hard copy is to be submitted.

1. [**10 points**] Write a pseudo-code of a procedure to check whether a given graph in adjacency list format is undirected.

2. [**10 points**] A bipartite graph is a graph $G = (V, E)$ whose vertices can be partitioned into two sets V_1 and V_2 ($V_1 \cup V_2 = V$ and $V_1 \cap V_2 = \emptyset$) such that there are no edges between vertices in the same set, i.e., if $(u, v) \in E$, then u and v must be from different sets. Give a linear time algorithm to determine if a given graph is bipartite.

3. [**10 points**] The reverse of a directed graph $G = (V; E)$ is another directed graph $G^R = (V, E^R)$, on the same vertex set V but all edges reversed, i.e. $E^R = \{(v, u) :$

$(u, v) \in E\}$. Give a linear-time algorithm to find the reverse of a graph in adjacency list format.

4. [**15 points**] Suppose a computer science curriculum consists of n courses, all of them mandatory. The prerequisite graph has a node for each course and an directed edge from node u to node v if u is a prerequisite of v. Find a linear time algorithm to compute the minimum number of semesters necessary to complete the curriculum, assuming that a student can take an unlimited number of courses in one semester.

5. [**20 points**] Draw a graph with at least two negative weight edges for which Dijkstra's algorithm produces the wrong answer. Draw another graph with at least one negative weight edge for which Dijkstra's algorithm produces the correct answer. (Try to draw small graphs and also tell the source node so that your solution can be easily verified.)

6. [**25 points**] Describe a linear time algorithm to find shortest paths in directed acyclic graphs. Prove the correctness of the algorithm.

7. [**30 points**] Suppose we have a directed graph $G = (V; E)$ describing a computer network, where vertices correspond to hosts or routers and edges correspond to network links. Also assume that for each network link $(u, v) \in E$, we are given a measure $r(u, v)$ of the reliability of this link: specifically, $r(u, v) =$ the probability that a packet sent across the link (u, v) will not be lost while it is transiting that link. You may assume that these probabilities are independent. So, if we have a path $(v_0, v_1), (v_1, v_2), \ldots, (v_{k-1}, v_k)$, the probability that a packet sent along this path makes it from v_0 to v_k successfully is given by $r(v_0, v_1) \times r(v_1, v_2) \times \cdots r(v_{k-1}, v_k)$. Given the graph G, the reliability measure $r(.,.)$, and vertices $s, t \in V$, your job is to find a path from s to t of maximum reliability. Design an efficient algorithm to solve this problem.