Intro to Algorithms CSCI 2300, Homework 5

Reading assignment: read chapter 3

Video assignment: watch the video of lecture 9,10, 11.

Q1. (10 points) An undirected graph is said to be bipartite if all its vertices can be partitioned into two disjoint subsets X and Y so that every edge connects a vertex in X with a vertex in Y. Design a linear time, i.e., O(|V| + |E|), time algorithm to check if a graph is bipartite or not.

Q2. (15 points) Answer the following questions:
a. Prove that a non-empty DAG must have at least one source.
b. What is the time complexity of finding a source in a directed graph or to determine such a source does not exist if the graph is represented by its adjacency matrix? Describe the algorithm.
c. What is the time complexity of finding a source in a directed graph or to determine such a source does not exist if the graph is represented by its adjacency list? Describe the algorithm.

Q3. (10 points) Describe a linear time algorithm to compute the neighbor degree for each vertex in an undirected graph. The neighbor degree of a node x is defined as the sum of the degree of all of its neighbors.

Q4. (20 points) Consider a directed graph that has a weight w(v) on each vertex v. Define the reachability weight of vertex v as follows:

$$r(v)=\max\{w(u)|u \text{ is reachable from } v\}$$

That is, the reachability weight of v is the largest weight that can be reached from v. Answer the following questions:
a. [10pnts] Assume the graph is a DAG. Describe a linear time algorithm to compute the reachability weight for all vertices.

b. [10pnts] Assume that the graph is a general directed graph (with possible cycles). Describe a linear time algorithm to find the reachability weight for all vertices.