CSC-421 Applied Algorithms and Structures
Winter 2019

Instructor: Iyad Kanj
Office: CDM 832
Phone: (312) 362-5558
Email: `ikanj@cs.depaul.edu`
Office Hours: Monday & Wednesday 4:00-5:30
Course Website: https://d2l.depaul.edu/

Assignment #4
(Due March 18)

1. (10 points) Textbook, page 662, exercise 24.3-1.

2. (60 points) Implement Dijkstra's algorithm for computing a shortest path from a designated vertex (A) to a designated vertex (B) in a directed graph. Your implementation should use a minimum heap as a supporting data structure. Please follow the instructions below.

   - **Programming languages.** You can use any standard programming language such as **C**, **C++**, **Visual C++**, **C#**, **Java**, or **Python**.
   - **Program requirements.** Your program should read the graph from a text file (assumed to be in the same folder as the source code) that **has the same format** as the input files uploaded on D2L (in the same folder as the assignment). Each input file starts with a line containing the number of vertices in the graph. The vertices are assumed to be numbered alphabetically starting with vertex A. Each subsequent line in the input file contains the tail of an edge followed by a space, the head of the edge followed by a space, and the weight of the edge, respectively. Your program should output the weight of a shortest path from vertex A to vertex B in the graph and the sequence of vertices on a shortest path from A to B; the format of the output should match the format of the solution files uploaded on D2L (same folder).

- **Material to be submitted.** The files containing your source code. Make sure that the files compile and run. The grader will test your programs on the uploaded test files (text files). So make sure that your programs run on the uploaded files.

3. (10 points) An *independent set* in a graph G is a set of vertices I in G such that no two vertices in I are adjacent (neighbors). The *maximum independent set problem* is, given a graph G, to compute an independent set of maximum size (maximum number of vertices) in G. Pinocchio claims that he has a greedy algorithm that solves the maximum independent set problem. Pinocchio's algorithm works as follows. The algorithm initializes the set I to the empty set, and repeats the following steps: Pick a vertex in the graph with the minimum degree, add it to the set I, and remove it and all the vertices adjacent to it from the graph. The algorithm stops when the graph is empty. Does Pinocchio's greedy algorithm always produces a maximum independent set? Prove your answer (if it does, give a proof; if it does not, give a counter example, that is, a graph on which Pinocchio's algorithm does not produce a maximum independent set).

4. (10 points) Textbook, page 422, exercise 16.1-3.

5. (10 points) Textbook, pages 637-638, exercise 23.2-8.

Please create a single ".zip" file containing all your answers to the problems in this assignment (i.e., source code as well as your solutions to the other problems) and upload it on D2L. Your solutions to problems 1, 3, 4, 5 should be provided as a separate document from your source code for problem 2, but in the same zipped file.