CS5800 Algorithms Spring 2015 Assignment #4

1. (25 points) For the following directed graph compute the shortest paths from vertex 3 using the Leyzorek algorithm and a binary heap:

Edge	Length	Edge	Length	Edge	Length	Edge	Length
(1,2)	3	(1,7)	4	(2,3)	4	(2,9)	5
(3,10)	1	(4,2)	7	(6,4)	9	(6,7)	2
(6,9)	4	(7,4)	8	(7,6)	9	(7,10)	9
(8,6)	3	(8,9)	3	(8,10)	8	(9,3)	6
(9,7)	10	(10,3)	8	(10,4)	9	(10,8)	4

For each step show the distance that was computed and the binary heap. Note that the initial vertex is 3, not 1.

2. (25 points) For the following directed graph compute the shortest paths from vertex 3 using the Moore algorithm:

Edge	Length	Edge	Length	Edge	Length	Edge	Length
(1,3)	4	(1,4)	10	(1,5)	9	(2,6)	2
(2,7)	2	(3,7)	5	(3,8)	8	(4,5)	3
(4,9)	-1	(6,7)	4	(6,9)	-1	(7,4)	5
(7,5)	6	(7,6)	3	(8,9)	5	(8,10)	4
(9,8)	5	(10,4)	7	(10,6)	4	(10,8)	5

Show your answer using the table notation in the textbook. Do not show the steps performed during the execution of the algorithm.

3. (25 points) A company has a collection of tasks that must be performed. Each task is dependent on other tasks and requires a time to complete measured in days. There are no pairs of tasks that depend on each other. There is an initial (source) task and a final task. Each task has a list of persons who are associated with the task. Each list has at most 3 persons. A route through the tasks is a sequence of tasks each of which is dependent on the previous task in the route. The time requirement of a route is the sum of the times required to complete the tasks in the route. The company needs to know the routes from the source task to the final task that require the longest time, and for each longest route it needs the persons who are associated with the tasks of the route. It is not necessary to eliminate duplicates in a list of persons. Develop an algorithm for computing the routes from the source task to the final task that require the longest time and the persons who are associated with the tasks of the longest routes. Prove that your algorithm has linear logarithmic (i.e., n log n) time complexity.

4. (25 points) Topologically sort the following DAG using DFS:

Edge	Length	Edge	Length	Edge	Length	Edge	Length
(1,10)	-9.2	(2,10)	-2.0	(3,1)	-4.2	(3,2)	3.0

(3,4)	9.3	(3,7)	14.4	(3,9)	18.7	(3,10)	5.2
(4,2)	11.4	(4,6)	3.5	(5,2)	-0.7	(5,7)	-3.6
(5,10)	4.2	(6,7)	1.2	(6,10)	-3.6	(7,10)	-5.1
(8,1)	-1.0	(8,10)	-4.5	(9,5)	17.5	(9,8)	19.8

Compute the shortest paths to all other vertexes, starting at the first vertex in topological sorted order. Show each step in the DAG single-source shortest path algorithm. For each step show the distance and parent pointer that was computed.