CSI2114 - Spring 2006 - Assignment # 4

Due: July 12, 2006

Exercise 1. [10 points]

Suppose that graph *G* has the following adjacency lists:

1 - (2*;* 3*;* 4)
2 - (1*;* 5)
3 - (1*;* 4*;* 5; 6)
4 - (1*;* 3*;* 6)
5 - (2; 3; 6*;* 7*;* 8)
6 - (3; 4*;* 5*;* 9)
7 - (5*;* 8*;* 9)
8 - (5*;* 7)
9 - (6; 7*;* 8)

a) Draw *G*.

b) Give the sequence of vertices visited using depth-first search starting at vertex 1.

c) Give the sequence of vertices visited using breadth-first search starting at vertex 1.

Exercise 2. [10 points]

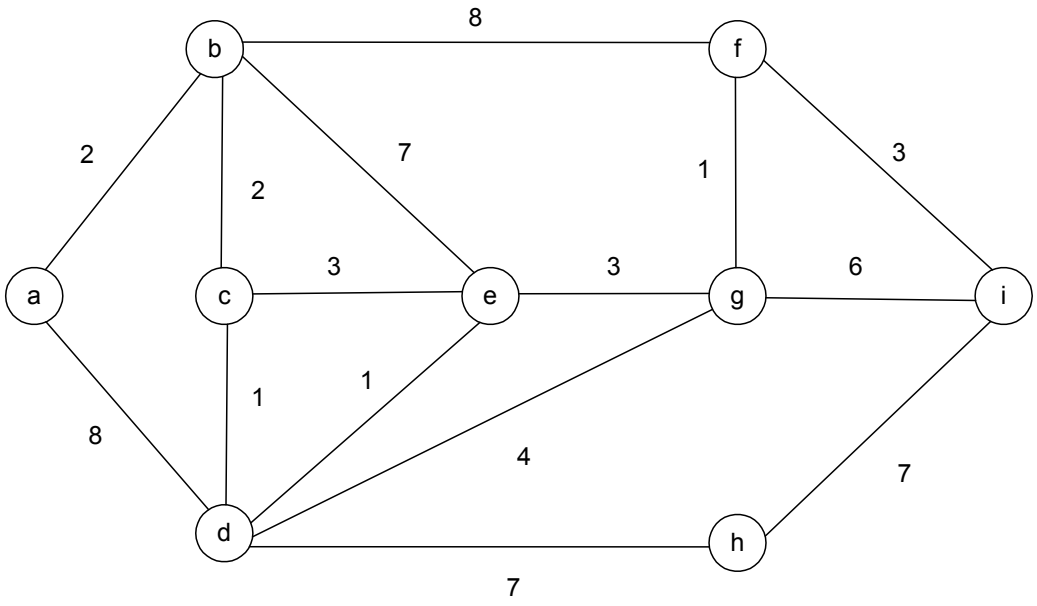
Consider the following greedy strategy for finding shortest path from vertex *start* to vertex *goal* in a given connected graph.

1. Initialize *path* to *start*.
2. Initialize *VisitedVertices* to {*start*}.
3. If *start* = *goal*, return *path* and exit; otherwise, continue.
4. Find the edge (*start, v*) of minimum weight such that *v* is adjacent to *start* and *v* is not in *VisitedVertices*.
5. Add *v* to *path*.
6. Add *v* to *VisitedVertices*.
7. Set *start* equal to *v* and go to step 3.

Does this greedy strategy always find a shortest path from *start* to *goal*? Either explain intuitively why it works, or give a counter example.

Exercise 3. [15 points]

Consider the following graph (weights are shown on the edges). Showing <u>all</u> intermediate steps,



a) Find the minimum spanning tree using Kruskal's algorithm.
b) Find the minimum spanning tree using Prim's algorithm.
c) Find the shortest path from *a* to *i* using Dijkstra's algorithm.

Exercise 4. [15 points]

Consider the sequence of keys {2, 12, 16, 15, 8, 18, 14, 6, 21, 7, 20, 5}. Sort the keys (showing <u>all</u> the intermediate steps) using:

a) Bubblesort
b) Merge Sort
c) Quick Sort