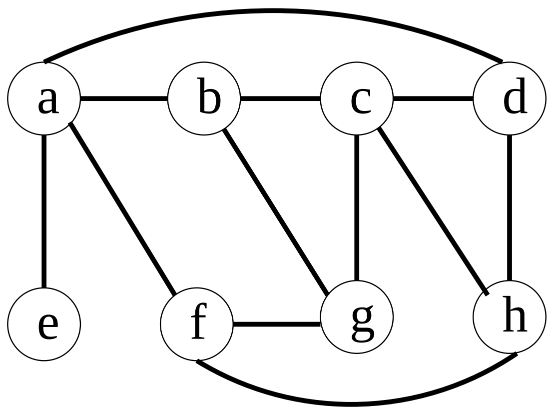
CMPS 2200 Introduction to Algorithms – Fall 14

10/28/14

6. Homework

Due **11/6/14** at the beginning of class

Remember, you are allowed to turn in homeworks in groups of two.



1. **Graph representation (7 points)**
   Assume that vertices are ordered alphabetically.
   (a) (2 points) Give the adjacency matrix representation for the graph above.
   (b) (2 points) Give the adjacency lists representation for the graph above.
   (c) (3 points) Give pseudo-code to convert a graph given in adjacency lists representation to its adjacency matrix representation. What is the runtime?

2. **Traversals (11 points)**
   Consider traversing the graph above, starting at vertex a. Assume the graph is given in your adjacency lists representation from question 1. Mark the results of the questions below in a copy of the graph.
   (a) Consider a depth-first traversal.
      i. (2 point) Give the discover time (d-value) and the finish time (f-value) of each vertex.
      ii. (1 point) Draw the depth-first tree.
      iii. (1 point) Mark each edge with its DFS classification (tree edge, back edge, forward edge, cross edge).

Flip over to back page ⟹

(b) Consider a breadth-first traversal.

   i. (2 points) Give the visit time stamp for each vertex (according to the pseudo code on slide 7).

   ii. (1 points) Draw the breadth-first tree.

(c) (4 points) Both DFS and BFS include the following `for`-loop referring to vertices `v` and `w`:

```
for each w adjacent to v do{
   // some statement that takes O(1) time
}
```

Give pseudo-code that implements this loop using (i) adjacency lists and (ii) an adjacency matrix. Analyze the runtime for both, assuming that the statement inside the loop takes O(1) time.

3. **Points on the line (6 points)**

Given a sequence $A = \{a_1, \ldots, a_n\}$ of points on the real line, sorted in non-decreasing order. The task is to determine the smallest set of unit-length (closed) intervals that contains all of the input points. Consider the following two greedy approaches:

(a) Let I be an interval that covers the most points in A. Add I to the solution, remove the points covered by I from A, and recurse/continue.

(b) Add the interval $I = [a_1, a_1 + 1]$ to the solution, remove the points covered by I from A, and recurse/continue.

One of these approaches is correct, the other one is not. Show which of the approaches is not correct by finding a counter-example. (The counter example consists of an example input and two “solutions” – one is the actual optimal solution, the other is the solution computed by the greedy algorithm, which is not as good as the optimal solution.)