COT 5407 Introduction to Algorithms
Homework 3

Due *in class* on Tuesday, October 30, 2017

This homework covers Ch 11, 12, 22, 23, 24.

---

1. **[5 points]**

   Demonstrate what happens when we insert the following keys 5, 28, 37, 15, 20, 33, 12, 17, 30 into a hash table with collisions resolved by chaining. Let the table have 9 slots, and let the hash function be $h(k) = k \ mod \ 9$.

2. **[10 points]**

   Suppose that you use a hash function h to hash n distinct keys into an array T that has length m. Assuming simple uniform hashing, what is the expected number of collisions? More formally, what is the expected cardinality of $\{\{k, l\} : k \neq l$ and $h(k) = h(l)\}$?

3. **[5 points]** Solve exercise 11.4-1 from Cormen.

4. **[5 points]**

   For the set of $\{1, 3, 6, 11, 16, 19, 21\}$ of keys, draw binary search trees of heights 2, 3, 4, 5, and 6.

5. **[5 points]**

   Briefly mention what is the difference between the binary-search-tree property and the min-heap property. Is it possible to use the min-heap property print out the keys of an tree with n nodes in $O(n)$? Justify your answer.

6. (**5 points**) Show that if a node in a binary search tree has two children, then its successor has no left child and its predecessor has no right child

7. (**10 points**) Hash Tables

   Consider a hash table that uses open addressing with a linear probing collision resolution mechanism, with table size $m = 7$ and the hash function:
   $h(k, i) = k + i \ (mod\ 7)$,
   here we assume that the key k is an integer.

   - Draw the hash table (with the above table size and hash function) that would be produced by inserting the following values, in the given order (from left to right), into an initially empty table:
     26, 32, 19, 11,3
   - Give the pseudocode for an algorithm to delete a given key from a hash table T. The hash table T uses open addressing and linear probing using the hash function described above.

8. **[5 points]** Solve exercise 12.3-3 from Cormen.

9. **[5 points]** Consider two graphs with positive weights. $G = (V, E, w)$ and $G' = (V, E, w')$ with the same vertices V and edges E such that for any edge $e \in E$, we have $w(e) = w(e)^2$. Prove or disprove the following statement: For any two vertices $u, v \in V$, any shortest path between u and v in G is also a shortest path in G'.

10. **[5 points]** Solve exercise 22.1-5 in Cormen

11. **[10 points]** DFS

    Draw the DFS tree of the graph in the figure above that will result if the first call to DFS is on vertex A and label each vertex with its discover ($v.d$) and finish ($v.f$) numbers. Draw tree edges as solid lines, and back, cross, and forward edges as dashed lines. Put a B, C, or F on each dashed line according to whether it is a back, cross, or forward edge.

12. **[5 points]** Solve exercise 22.4-1 in Cormen

13. **[5 points]** Solve exercise 22.5-2 in Cormen

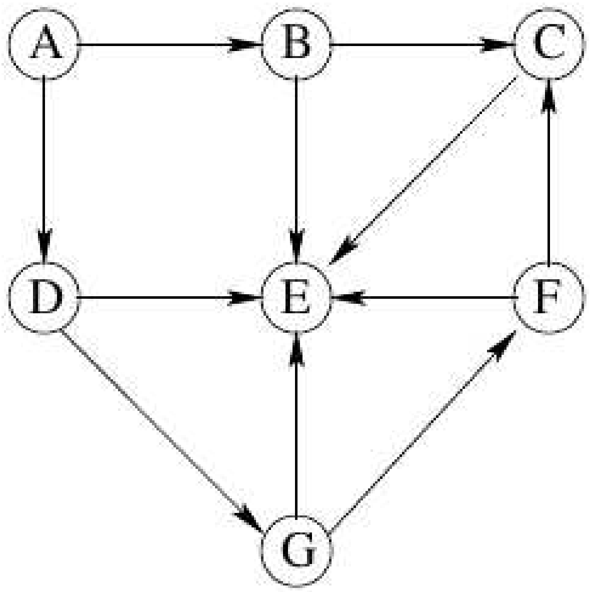

Figure 1. Assume that the adjacency lists are ordered so that vertices are ordered in alphabetical order in an adjacency list. For example, the adjacency list of A has the edge to B before the edge to D.

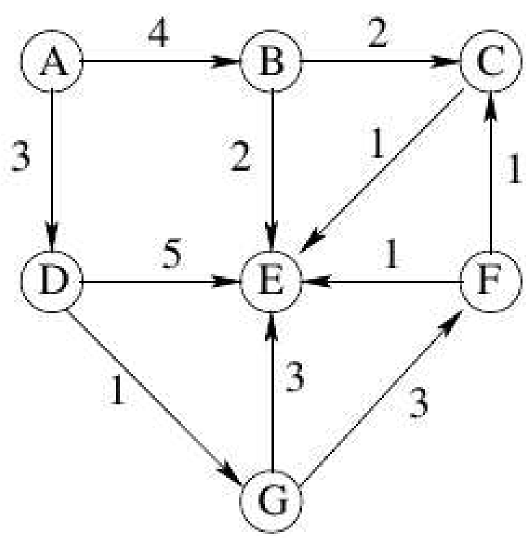

Figure 2. Graph for Dijkstra

14. **[10 points]** Dijkstra

    Execute Dijkstra's algorithm on the graph of the figure above starting at vertex A. If there are any ties, the vertex with the lower letter comes first.

    - List the vertices in the order in which they are deleted from the priority queue and for each the shortest distance from A to the vertex.

- Draw the shortest paths tree that results

15. (**10 points**) Kruskal and Prim

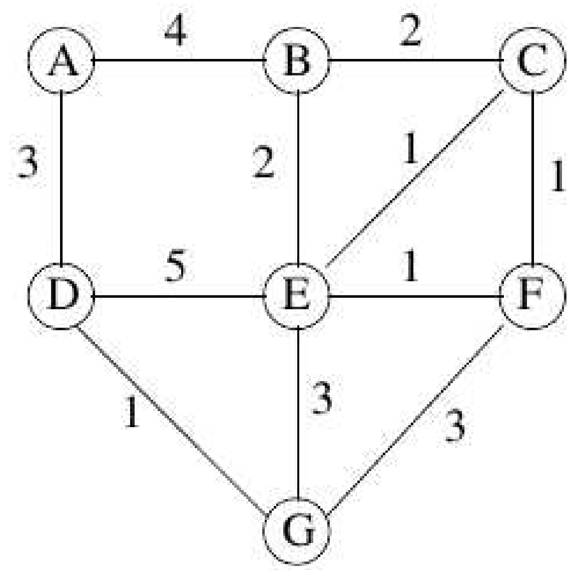

Figure 3. Graph for Kruskal

- Execute Prim's algorithm on the graph above starting at vertex A. If there are any ties, the vertex with the lower letter comes first. List the edges in the order in which they are added to the tree.
- Execute Kruskal's algorithm on the graph above starting at vertex A. Assume that equal weight edges are ordered lexicographically by the labels of their vertices assuming that the lower labeled vertex always comes first when specifying an edge, e.g. (C, E) is before (C, F) which in turn is before (D, G). List the edges in the order in which they are added to the developing forest.