University of Nevada, Las Vegas Computer Science 477/677 Spring 2015

Practice Examination for April 7, 2015

Finished

The entire practice examination is 410 points.

Name:____

No books, notes, scratch paper, or calculators. Use pen or pencil, any color. Use the rest of this page and the backs of the pages for scratch paper. If you need more scratch paper, it will be provided.

The examination may contain questions about topics covered on the homework assignments, even if those topics are not represented in the question below.

Some students have asked me whether "complexity questions" will be on the examination. Of course, you are expected to remember all material covered so far in the course, but there will not be any questions specifically related to the material covered before the first examination. What that means is, for example, I might ask you to find the recurrence relation which is appropriate for mergesort, since we covered mergesort since the last exam. I might also ask you the time complexity of something we covered since the last exam.

1. True or False. [5 points each]

   (a) _______ Linear probing is a method for resolving collisions in open addressing.

   (b) _______ The matrix representation for an n-node directed graph uses more rows and columns than a matrix representation for an n-node undirected graph.

   (c) _______ In open addressing, collisions in a hash tables can be addressed using chaining.

   (d) _______ A directed acyclic graph with more than two nodes has only one strongly connected component.

   (e) _______ A heap is a particular form of a search structure.

2. Fill in the blanks. [5 points each blank.]

   (a) The algorithm for computing the value of an expression in postfix notation uses a _______________. [Hint: it's a data structure.]

   (b) In a _____________________________ hash table, there are no collisions.

   (c) A graph is called _____________________________ if the number of edges is asymptotically less than n^2.

   (d) The number of edges of a connected 10-node undirected graph is no larger than than _______ and no smaller than _______. (Exact answer, no asymptotic notation).

   (e) If every strong component of a directed graph consists of a single point, then that directed graph is _____________________________.

(f) A directed graph is called ______________________________ ______________________________ if there is a directed path from any node to any other node.

(g) You should use a _______________ to implement depth-first search in a graph, but you should use a _______________ to implement breadth-first search. [Hint: data structures.]

3. Write pseudocode for binary search in an ordered array A. To save space, assume that A is already available, and do not declare any variables.

4. Consider the following sequence of values: 6, 7, -1, -2, 9, 6, 0, -2.

(a) Use the mergesort algorithm to sort the items in increasing order. Show the intermediate arrays. (Half a page is more than enough.) What is the total number of comparisons done by mergesort for this sequence?

(b) In the sorted sequence obtained above, if we apply binary search to search for value 1, how many comparisons are needed to decide whether 1 is part of the sorted sequence or not? (Use the pseudocode you wrote for Problem 3.)

5. [30 points] The bit-matrix below represents a directed graph G.

$$\begin{bmatrix} 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}$$

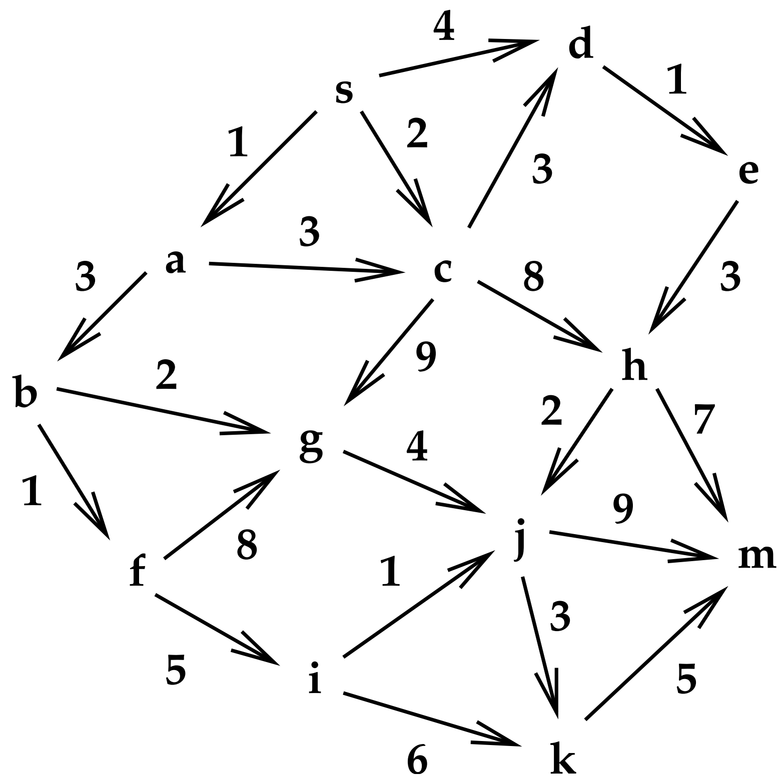
(a) Draw G.

(b) How many strong components does G have? ______ Circle the strong components in your drawing.

(c) Write the in-neighborlist for this graph.

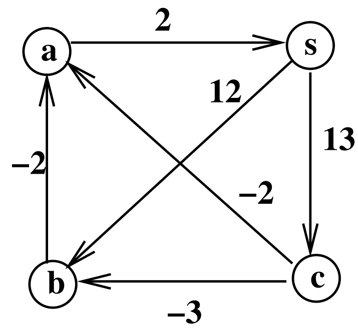
6. [20 points] Give an algorithm for computing the out-neighbor list of a directed graph, given the in-neighborlist. Do not write C++ or Java code.

7. [30 points] Consider the weighted directed graph G below, where the source vertex is **a**.



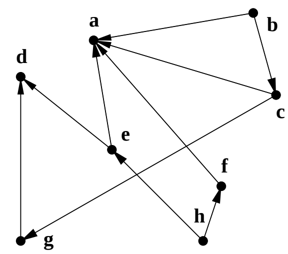
Use Dijkstra's algorithm to solve the single-source minimum path problem for G. You do not have to give code: just show the arrays V and *back*, as well as the contents of the heap, at each step.

8. [30 points] Consider the weighted directed graph below, where the source vertex is **s**.



Use Bellman-Ford algorithm to find solve the single-source minimum path problem, where the source is S. (Note that because some weights are negative, Dijkstra's algorithm does not work.) You should show the array V after one iteration of the main loop, then after two iterations, then after three iterations.

9. [10 points] Consider the directed acyclic graph below. Write the nodes of that graph in topological



order.

10. [30 points] Write the pseudocode for topological sort of the nodes of an acyclic directed graph. Assume that the names of the nodes are 1, 2, ... n.

11. [20 points] Write pseudocode for the Floyd-Warshall algorithm.

12. Consider the following array representing a weighted directed graph G.

$$\begin{bmatrix} \infty & -2 & \infty & 1 & \infty \\ \infty & \infty & 3 & \infty & \infty \\ 2 & \infty & \infty & -1 & \infty \\ \infty & \infty & \infty & \infty & 2 \\ 0 & \infty & \infty & \infty & \infty \end{bmatrix}$$

(a) [10 points] Draw a picture of G.

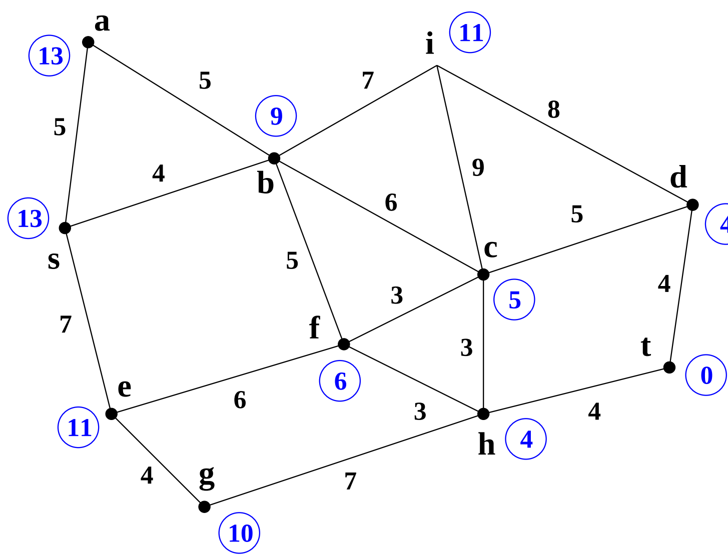
(b) [20 points] Draw the array V after 4 iterations of the Floyd-Warshall algorithm for the directed graph G.

13. [30 points] Let A be a 3-dimensional array, where $A[i, j, k]$ is defined if $i \in [0 \dots 5]$, $j \in [-3 \dots 8]$, and $k \in [1 \dots 9]$. Each entry of A takes three memory locations, and the entries are stored in column-major order, with base address 1024. What are the three addresses used to store $A[3, 2, 6]$?

14. Use the A* algorithm to find the shortest path from **s** to **t** in the weighted graph G shown below. Use the heuristic which is indicated by the circled numbers.

G can be considered to be a directed graph, where each edge is replaced by two directed edges, one in each direction, both the the same weight. Thus, the directed version of G has 26 directed edges.

Adjust each directed edge, using the give heuristic, and show the modified weighted graph. Then, find the shortest path from **s** to **t** using the modified weights.



15. [40 points] Use dynamic programming to find the maximum length monotone increasing subsequence of the following sequence:

$$5, 1, 3, 6, 9, 4, 7, 2, 8$$

(a) Identify the subproblems, and write the list of subproblems in topological order.

(b) Compute the solution to each subproblem. (Just state the answer; it is not necessary to explain the computation.)

(c) Write the maximum length monotone increasing subsequence.

The internet should give you some clues. For example, http://www.codechef.com/wiki/tutorial-dynamic-programming contains pseudo-code for an $O(n^2)$-time algorithm for this problem, but it has an error. Can you find that error?