CS 483 Analysis of Algorithms - Spring 2016

Midterm - March 21, 2016

Professor: Jana Kosecka

Student's name:

- Write your name in the space provided above
- This exam contains six problems. You have 75 minutes to earn 70 points
- Write your solutions in the space provided
- Do not waste time re-deriving facts that we have studied. It is sufficient to cite known results
- Do not spend too much time on any one problem. Read all the problems first, and attack them in the order that allows you to make the most progress
- Show your work, as partial credit will be given. You will be graded not only on the correctness of your answer, but also on the clarity with which you express it. Be neat.
- Good luck!

Question	Score	Points
1	10	
2	15	
3	10	
4	10	
5	10	
6	10	
7	5	
Total	70	

1. True/False Questions [10 pts]

For each of the following propositions, determine whether it is **(T)**RUE or **(F)**ALSE. No explanation is needed. Incorrect answers or unanswered questions are worth 0 points.

f and g are asymptotically nonnegative functions.

1. $f(n) = O(g(n))$ and $g(n) = O(f(n))$ implies that $f(n) = \Theta(g(n))$. **T** **F**

2. $\frac{n}{log(n)} = \Omega((log(n))^{log(n)})$ **T** **F**

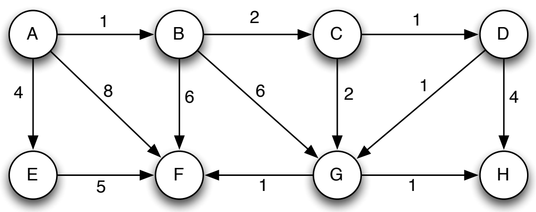
3. $10log(n) = \Theta(log(n^2))$ **T** **F**

4. If $f(n) = O(g(n))$ then $3^{f(n)} = O(3^{g(n)})$ **T** **F**

5. If $f(n) = \Omega(g(n))$ then $g(n) = O(f(n))$ **T** **F**

2. Shortest Paths [15 pts]

Use Dijkstra's algorithm to find the shortest distances from node A to all the other nodes. For each iteration of the algorithm, show the set of explored nodes so far, and draw a table with the intermediate distance values of all the nodes. Show the final shortest-path tree.



3. Graph Representation [10 pts]

Which graph representation (adjacency matrix or adjacency list) is more efficient for the following operations? Describe the worst-case running time of each operation for both data structures. Assume the graph is undirected.

1. Finding whether there is an edge between two nodes u and v.

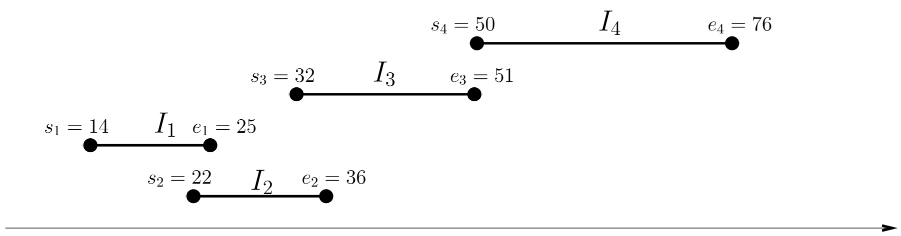
2. Computing the degree of a node.

4. Connectivity in Directed Graphs [10 pts]

Let G be a directed graph. Provide an algorithm to decide whether G is strongly connected or not. What is the worst-case time complexity of your algorithm? Motivate your answer.

Problem 5 [10 pts]

Consider a set of intervals $I = \{I_1, I_2, \ldots, I_n\}$, where $I_i = (s_i, e_i)$ is a pair of real numbers s_i and e_i such that $s_i < e_i$. An example is given below. Two intervals I_i and I_j do not overlap if $e_j < s_i$ or $e_i < s_j$. Moreover, we define the length of an interval as $|I_i| = e_i - s_i$, and the length of a set I of intervals as $|I| = \sum_{I_i \in I} |I_i| = \sum_{I_i \in I}(e_i - s_i)$. In the *interval scheduling problem*, we are asked to find a subset I' of I such that no intervals in I' overlap, and the length of I' is maximized.



(a) Develop a greedy algorithm to solve the interval scheduling problem for a given I.

(b) Provide an example to show that your greedy algorithm is not optimal.

Problem 6 [10 pts]

You are given a sorted array of numbers where every value except one appears exactly twice; the remaining value appears only once. Design an efficient algorithm for finding which value appears only once and determine its running time. (e.g. $1, 1, 2, 2, 3, 4, 4, 6, 6, 8, 8, 10, 10$). Trivial algorithm will take $O(n)$ steps and will just traverse the array once.

6. Extra Credit! [5 points]

List things you liked about the class so far and things you would like to see changed. There is no *a-priori* defined correct answer to this question. Say what you think and you'll receive the credit. It will help me improve future offerings of the course.