NAME: _____

CSE 331

Introduction to Algorithm Analysis and Design

Sample Mid-term Exam-I: Fall 2016

Atri Rudra

DIRECTIONS:

- Closed Book, Closed Notes except for one $8\frac{1}{2}" \times 11"$ review sheet.
- Time Limit: 50 minutes.
- Answer the problems on the exam paper.
- Make sure you write your NAME on the paper.
- If you need extra space use the back of a page.

1a	/5
1b	/5
1c	/5
1d	/5
1e	/5
1f	/5
1g	/5
1h	/5
Total	/40

FEW GENTLE REMINDERS:

- You can quote any result that we covered in class or any problem that was there in a homework (but remember to explicitly state where you are quoting a result from).
- If you get stuck on some problem for a long time, move on to the next one.
- The ordering of the problems is somewhat related to their relative difficulty. However, the order might be different for you!
- You might be better off by first reading all questions and answering them in the order of what you think is the easiest to the hardest problem. Keep the points distribution in mind when deciding how much time to spend on each problem.

1. ($8 \times 5 = 40$ points) Answer True or False to the following questions and briefly JUSTIFY each answer. A correct answer with no or totally incorrect justification will get you 2 out of the total 5 points. An incorrect answer (irrespective of justification) will get you **no points**. (Recall that a statement is true only if it is logically true in all cases while it is is false if it is not true in some case).

   (a) For any instance of the stable marriage problem with n men and n women, there are $n! = n \times (n-1) \times \ldots \times 1$ many possible perfect matchings.

   (b) Consider $f(n) = \log \log n$ and $g(n) = 10^{10^{10^{10^{10}}}}$. Then $f(n)$ is $O(g(n))$.

   (c) Consider $f(n) = n^n$ and $g(n) = 2^{400n}$. Then $f(n)$ is $\Omega(g(n))$.

   (d) There is an algorithm that sorts n numbers $a_1, \ldots, a_n$ where $a_i \in \{0, 1\}$ for each $i \in [n]$ in $O(n)$ time.

(e) BFS is a linear time algorithm. (Recall that an algorithm is a linear time algorithm if it runs in time $O(N)$ on inputs of size N.)

(f) For any graph, there is a unique BFS tree for it.

(g) Every directed graph on n vertices with at least $n-1$ edges is strongly connected.

(h) Given a graph on n vertices in its adjacency matrix, there is an $O(n^2)$ time algorithm to convert it into its adjacency list representations.