University of Washington
Department of Computer Science and Engineering
CSEP 521, Winter 2013

February 2, 2013

Homework 5, Due Monday, February 11, 2013

Note: these problems come from old midterm exams, and should be solvable in substantially less time than the typical homework problems.

Problem 1 (10 points):

Consider the stable matching problem.

a) Given sets M and W with $|M| = |W| = n$, describe a set of preference lists for the elements of M and W such that the stable matching problem for M and W has a unique solution.

b) Prove that your instance from part a) has a unique stable matching. You may use the following definition of stable matching: For matching with m matched to w and m' matched to w', (m, w') is an instability if m prefers w' to w and w' prefers m to m'. A M matching is said to be *stable* if it has no instabilities.

Problem 2 (10 points):

Let $G = (V, E)$ be an undirected graph.

a) True or false: If G is a tree, then G is bipartite. Justify your answer.

b) True or false: If G is not bipartite, then the shortest cycle in G has odd length. Justify your answer.

Problem 3 (10 points):

Let $G = (V, E)$ be a directed graph with n vertices.

a) True or false: If G has at least n edges, then G has a cycle. Justify your answer.

b) True or false: If every vertex of G has out degree at least one, then G has a cycle. Justify your answer.

Problem 4 (10 points):

Let $G = (V, E)$ be an undirected graph with edge weights. Assume that all edge weights are distinct.

a) True or false: If e is the minimum weight edge, e is in the minimum spanning tree. Justify your answer.

b) True or false: If e is the maximum weight edge, e cannot be in the minimum spanning tree. Justify your answer.

Problem 5 (10 points):

Give solutions to the following recurrences. Justify your answers.

a)

$$T(n) = \begin{cases} T(\frac{n}{4}) + n & \text{if } n > 1 \\ 1 & \text{if } n \leq 1 \end{cases}$$

b)

$$T(n) = \begin{cases} 9T(\frac{n}{3}) + n^2 & \text{if } n > 1 \\ 1 & \text{if } n \leq 1 \end{cases}$$

Problem 6 (10 points):

The sequence $A = a_1 a_2 \ldots a_n$ is a subsequence of $B = b_1 b_2 \ldots b_m$ if the elements of A occur in order in B, or more formally, if $a_1 = b_{i_1}, a_2 = b_{i_2}, \ldots, a_n = b_{i_n}$ for $i_1 < i_2 < \cdots < i_n$.

Give an $O(n + m)$ time algorithm to test if A is a subsequence of B. Justify that your algorithm is correct and that it satisfies the run time bound.