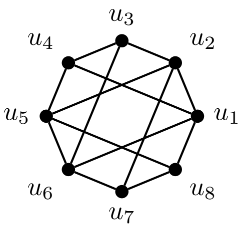
Discrete Mathematics

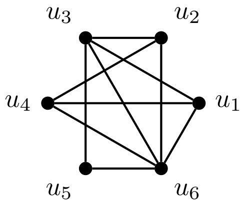
Assignment 12

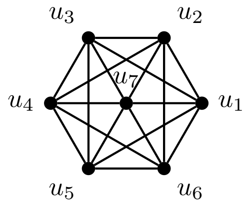
Date Due: 8:00 PM, Monday, the 8th of August 2011

Office hours: Tuesdays, 12:00-2:00 PM, and Wednesdays, 12:00-1:00 PM

Exercise 1. Determine whether each given graphs is planar. Eitehr draw an isomorphic graph without crossing edges, or prove that the graph is non-planar.

i) 

ii) 

iii) 

(3 × 2 Marks)

Exercise 2. Find the chromatic number for the three graphs in Exercise 1.
(3 × 1 Mark)

Exercise 3. Show that the Marriage Condition 3.2.74 is equivalent to the Marriage Condition 3.2.78.
(4 Marks)

Exercise 4. Prove the Harem Theorem 3.2.80.
(3 Marks)

Exercise 5.

i) Show that a simple graph is a tree if and only if it is connected, but the deletion of any of its edges produces a graph that is not connected.

ii) Let G be a simple graph with n vertices. Show that G is a tree if and only if G is connected and has $n-1$ edges.

(3 + 3 Marks)

Exercise 6. Show that a full m-ary balanced tree of height h has more than m^{h-1} leaves. Deduce the second statement of Corollary 3.3.16 regarding the height of full and balanced trees.
(3 Marks)

Exercise 7. One of four coins may be counterfeit. If it is counterfeit, it may be lighter or heavier than the others. How many weighings are needed, using a balance scale, to determine whether there is a counterfeit coin, and if there is, whether it is lighter or heavier than the others? Describe an algorithm to find the counterfeit coin (or establish its non-existence) and determine whether it is lighter or heavier using this number of weighings.
(4 Marks)

Exercise 8. The *tournament sort* is a sorting algorithm that works by building an ordered binary tree. We represent the elements to be sorted by vertices that will become the leaves. We build up the tree one level at a time as we would construct the tree representing the winners of matches in a tournament.

Working left to right, we compare pairs of consecutive elements, adding a parent vertex labeled with the larger of the two elements under comparison. We make similar comparisons between labels of vertices at each level until we reach the root of the tree that is labeled with the largest element.

The tree constructed by the tournament sort of 22, 8, 14, 17, 3, 9, 27, 11 is illustrated in part (a) of the figure at right. Once the largest element has been determined, the leaf with this label is relabeled by $-\infty$, which is defined to be less than every element. The labels of all vertices on the path from this vertex up to the root of the tree are recalculated, as shown in part (b) of the figure. This produces the second largest element. This process continues until the entire list has been sorted.

i) Use the tournament sort to sort the list 17,4, 1, 5, 13, 10, 14, 6. Draw the tree at each step.

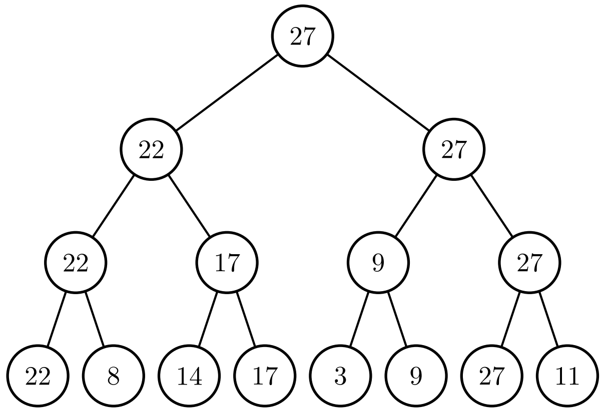
ii) Assuming that n, the number of elements to be sorted, equals 2^k for some positive integer k, determine the number of comparisons used by the tournament sort to find the largest element of the list using the tournament sort.

iii) How many comparisons does the tournament sort use to find the second largest, the third largest, and so on, up to the $(n-1)$st largest (or second smallest) element?

iv) Show that the tournament sort requires $\Theta(n \log n)$ comparisons to sort a list of n elements. [Hint: By inserting the appropriate number of dummy elements defined to be smaller than all integers, such as $-\infty$, assume that $n = 2^k$ for some positive integer k.]

(2 + 3 + 3 + 3 Marks)

(a) 27 is the largest element.



(b) 22 is the second largest element.