Dalhousie University Faculty of Computer Science
Design and Analysis of Algorithms
Assignment 6 CSCI 3110 Due: 15 Nov 2010

1. (a) Ex. 2.4 p. 71
   (b) Ex. 2.5(f) and (i)
   (c) Ex. 2.12 p. 73
2. (a) Ex. 2.25
   (b) Ex. 2.29
3. (a) Ex. 2.16 p. 73
   (b) Consider an $m \times n$ array A of integers. Each row and colum of the array are in ascending order: $A[i,j] \leq A[i,j+1], \quad i = 1 \ldots n, \quad j = 1 \ldots m-1$ and $A[i,j] \leq A[i+1,j], \quad i = 1 \ldots n-1 \quad j = 1 \ldots m$.
   Design an algorithm that determines if a given integer x is in A, if so, return it's position otherwise return a -1. Explain your algorithm with words and diagram(s). Show that your alogrithm terminates and find its worst case complexity. *Hint: Start by comparing x with $A[1,m]$*
4. (a) Draw the recursion tree to guess a solution and prove your guess for
   $$T(n) = T(n-a) + T(a) + cn$$
   for constants $a \geq 1$ and $c > 0$. Assume $T(n) = \Theta(1)$ for $n \leq a$.
   (b) Draw the recursion tree to guess a solution and prove your guess for
   $$T(n) = T(\alpha n) + T((1-\alpha)n) + cn$$
   for constants $0 < \alpha < 1$ and $c > 0$.
   (c) Suppose that the splits at each level of quicksort are in the proportion $1-\alpha$ to α where $0 < \alpha \leq 1/2$ is a constant. Show that the minimum depth of a leaf in the recursion tree is approx. $-\lg n/\lg \alpha$ and the maximum depth is approx $-\lg n/\lg(1-\alpha)$
5. (a) Give an algorithm that sorts an array of size n, only containing the elements 1,2 and 3, in linear time. Give a plausible argument that your algorithm sorts the array in linear time
   (b) Construct an example input for which quicksort will use $\Omega(n^2)$ comparisions when the pivot is chosen as the median of the first, last and middle elements of the sequence.