CSE 101 Homework 2

Due November 2, 2004

1. Consider the following pseudocode for DFS:

   1 Frontier <- { x }
   2 Visited[x] <- T
   3 forall y != x
   4     Visited[y] <- F
   5 while Frontier not empty
   6     y <- Pop(Frontier)
   7     for z in Adj(y)
   8         if Visited[z] = F
   9             Push(Frontier, z)
10         Visited[z] <- T

   (a) Describe how to modify the above code so that the algorithm returns “Yes” if the graph
       G contains a cycle and returns “No” if it doesn’t. Try to make as few modifications as
       possible. Explain why your solution works.
   (b) What is the running time of the modified algorithm? Explain.

2. Recall the skyline problem from lecture: given a list of buildings $b_1 = (l_1, r_1, h_1), \ldots, b_n =
   (l_n, r_n, h_n)$, output a list $(x_1, y_1), \ldots, (x_m, y_m)$ such that $x_1$ is the leftmost end of the skyline,
   $x_m$ is the rightmost end of the skyline, $x_1 < x_2 < \ldots < x_m$, and from $x_i$ to $x_{i+1}$ the skyline
   has height $y_i$ (where $y_m = 0$).

   (a) Give a divide and conquer algorithm for computing the skyline.
   (b) What is the running time of this algorithm? Explain.

3. (a) Draw all valid min-heaps (in tree form) for elements with the following priorities: 3,8,10,11,14.
   (b) Let $n = 2^k - 1$ for some $k \geq 1$. Let $T(n)$ be the number of different min-heaps for $n$
       elements with distinct priorities (two heaps can be different even if they are isomorphic).
       Write a recurrence relation for $T(n)$. Make sure to specify the value of $T(1)$.
   (c) Solve the above recurrence relation (hint: expand it out).

4. Consider a modified version of mergesort that recurses on the subarrays $A[1..n/4]$ and
   $A[n/4+1..n]$ instead of $A[1..n/2]$ and $A[n/2+1..n]$.

   (a) Write a recurrence for the worst-case running time of the modified mergesort.
   (b) Is the worst-case running time of modified mergesort worse than that of regular merge-
       sort? Prove it.