Analysis of Algorithms - Final

K. Subramani
LCSEE,
West Virginia University,
Morgantown, WV
{ksmani@csee.wvu.edu}

1 Instructions

1. The final is due on Monday, December 10 at 10 am.
2. Each question is worth 4 points.
3. Attempt as many problems as you can. You will be given partial credit, as per the policy discussed in class.

2 Problems

1. Consider the following algorithm to compute the Minimum Spanning Tree of an undirected graph G.

Function MST-COMPUTE ($G = \langle V, E, w \rangle$)
1: Order the edges in non-increasing order of their weights w.
2: $T \leftarrow E$.
3: **for** (each edge e processed in non-increasing order) **do**
4: **if** ($T - \{e\}$ is a connected graph) **then**
5: $T \leftarrow T - \{e\}$.
6: **end if**
7: **end for**
8: **return**(T).

Algorithm 2.1: A new MST Algorithm

   Is this algorithm correct? Justify your answer.

2. Argue that there exists a constant $c > 0$, such that any comparison-based sorting algorithm takes at least $c \cdot n \cdot \log n$ comparisons to sort an array of n numbers.
3. Let $X = \langle A, B, C, B, D, A, B \rangle$ and $Y = \langle B, D, C, A, B, A \rangle$. Compute the LCS of X and Y using the technique discussed in class. You are *required* to show at least some of the computations.
4. Let u be a vertex in a directed graph G with both an incoming edge and an outgoing edge. Is it possible for a DFS on G to produce a forest in which u is the only vertex?
5. Let T denote a binary tree representing the prefix character code of some alphabet. Argue that if T is not full, it cannot be optimal. Recall that a binary tree is full, if every internal node has exactly two children.