

CPSC 320: Intermediate Algorithm Design and Analysis
Assignment #3, due Thursday, February 9th, 2012 at 11:00

- [9] 1. In class, we discussed an algorithm that constructs a minimum prefix tree for a string of length n with m distinct characters. Show how to construct the minimum prefix tree in $O(m)$ time in the case where the input to the algorithm is a list of pairs

$$(c_1, f_1), (c_2, f_2), \dots, (c_m, f_m)$$

where f_i is the frequency of the character c_i , and the f_i 's are sorted in increasing order (that is, $f_1 \leq f_2 \leq \dots \leq f_m$). Hint: what can you say about the sequence of frequencies assigned to the nodes created when you merge two already existing trees?

- [13] 2. Explain what adjustments, if any, need to be made to Dijkstra's single-source shortest paths algorithm and/or to the given input graph to solve the following problems.
- a. Find a shortest path between two given vertices of a weighted graph or a directed graph (this variation is called the *single-pair shortest-path problem*).
 - b. Find the shortest path to a given vertex from each other vertex of a weighted directed graph (this variation is called the *single-destination shortest-paths problem*).
 - c. Solve the single-source shortest-paths problem in a graph with non-negative numbers assigned to its vertices (and the length of the path defined as the sum of the vertex numbers on the path).