CPSC 320 | Problem Set 6 | 6 November 2009

W. Evans | Due: 13 November 2009 at the start of class.

The grading policy for this homework is as follows: If you leave a question blank, you receive 1 point for that question. If you answer a question, the question will be graded on a scale from 0 to 5. This homework has four questions.
You do not need to rewrite the question or copy down pseudo-code that was presented in class.

1. The greedy scheduling algorithm to find a maximum size set of non-overlapping jobs is:

   GreedySched($S = [(s_1, f_1), (s_2, f_2), \ldots, (s_n, f_n)]$)
   1. Sort jobs so that $f_1 \leq f_2 \leq \cdots \leq f_n$.
   2. $A = \emptyset$
   3. For $i = 1$ to n
      If job i doesn't overlap any job in A then $A = A \cup \{i\}$
   4. Return A

   We can obtain three other scheduling algorithms by modifying step 1. For each of the following proposed replacements for step 1, either: write "works" if the resulting algorithm always produces an optimal (maximum size) schedule for input S (no proof is necessary), or give an input for which the resulting algorithm would fail to produce an optimal schedule.

   (a) 1. Sort jobs so that $f_1 \geq f_2 \geq \cdots \geq f_n$.

   (b) 1. Sort jobs so that $s_1 \leq s_2 \leq \cdots \leq s_n$.

   (c) 1. Sort jobs so that $s_1 \geq s_2 \geq \cdots \geq s_n$.

2. Suppose edge weights are integers in the set $\{0, 1, 2, \ldots, W\}$. How would you modify Dijkstra's algorithm to compute the shortest paths in $G = (V, E)$ from a given source vertex s to all other vertices in $O(Wn + m)$ time?

   Hint: How many different priorities are in the priority queue at any given time?