Algorithm Design Techniques - COMP 360B

MIDTERM EXAM Student ID: NAME:

March 4, 2003 10:05 - 11:20

All answers to be written on this question sheet, which has 4 pages.
You may use the back of the page if necessary.
No calculators, cell phones, books, notes or cheating.

(10 pts) 1. Let $G = (V, E, U)$ be a bipartite graph with vertex sets V and U, and edge set E. A **semi-matching** in the graph is a set of edges M such that no two edges in M share a common endpoint in V. (A matching is a set of edges with no common endpoint in either V or U). Each edge e has a weight $w(e)$. Prove that the greedy selection algorithm correctly computes a maximum weight semi-matching by:

(a) Stating three properties for selection problems to be solvable by greedy selection, and

(b) Showing how the semi-matching problem satisfies each of these properties.

(10 pts) 2. Consider a grid graph with vertices $(i, j), 0 \le i, j \le n$. Each vertex (i, j) with $i < n$ has a horizontal edge directed to $(i + 1, j)$ with a weight of $H[i, j]$. Each vertex (i, j) with $j < n$ has a vertical edge directed to $(i, j + 1)$ with a weight of $V[i, j]$.
The **min_weight** of a directed path P in the graph is the minimum edge weight that appears in P.
Write a dynamic programming pseudocode to find a path from (0,0) to (n, n) which has maximum min_weight, and runs in $O(n\,3)$ time.

(10 pts) 3. Let $G = (V, E)$ be a directed graph with weights assigned to each edge. Suppose there is one negative weight edge, and that all the other edge weights are positive. Explain how to use one application of **Dijkstra's algorithm** to determine if G has a negative weight cycle. Compare the time complexity of your method to the alternative approach of using the Bellman-Ford algorithm.

(10 pts) 4. Consider the bipartite graph $G = (V, E, U)$, with vertex sets $V = \{a, b, c, d\}$ and $U = \{e, f, g, h\}$, and edges $E = \{af, ah, bf, bg, bh, ce, cf, cg, ch, dg\}$.
Start from the matching consisting of edges af, bg, ch and find a maximum matching for this graph by using either the alternating paths algorithm, or by converting to a network and then using the augmenting paths algorithm.

Score: 1: 2: 3: 4: Total: