CS524

FINAL EXAM

Name______________

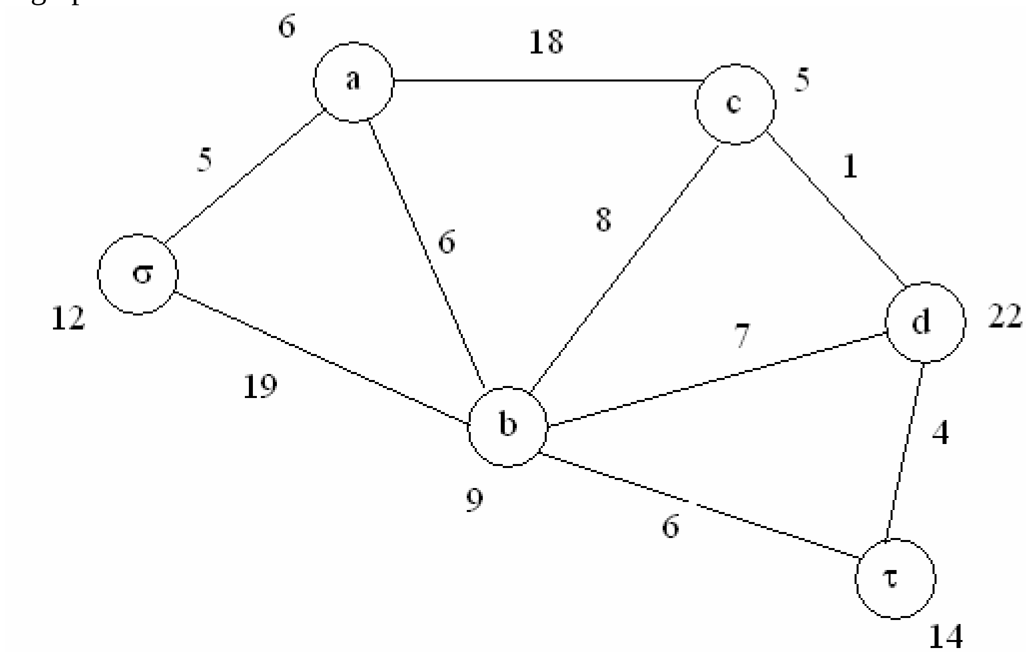
Date: December 14, 2005

All nonelectronic documentation permitted

1. (20 points) Describe an algorithm to test if graph $G = (V, E)$ is a forest. Your algorithm should have time complexity in $O(|V|)$. Beware: $O(|V|) \neq O(|V| + |E|)$.

2. (25 points) Suppose you are given a graph $G=(V,E)$, $\sigma,\tau\in V$, and a weight function $w:V\cup E\to\Re^{+}$. That is, there is a positive weight associated with each edge **and** each vertex of *G*. The *length* of a path is the sum of the weights of the vertices plus the sum of the weights of the edges of the path. Describe an algorithm with execution time in $O\left(|V|^2\right)$ to find a shortest path from σ to τ.

For graph



the shortest path from σ to τ is the path (σ,a,b,τ) of length

$$58=w(\sigma)+w(\sigma a)+w(a)+w(ab)+w(b)+w(b\tau)+w(\tau).$$

3. (25 points) Suppose you want to compute the distances between **all** pairs of vertices in a sparse graph $G = (V, E)$, where sparse means that $|E| = O(|V|)$. That is, if $V = \{1, \ldots, n\}$, you want to compute the values of $n \times n$ array *D*, where *D*[*i*,*j*] is the distance between vertex *i* and vertex *j*. You want to do this as efficiently as possibly. That is, you want to minimize the asymptotic time complexity. Would you use the FLOYD-WARSHALL ALGORITHM? Justify your response.

4. (30 points) An *independent set* of vertices in graph $G=(V,E)$ is a set $S \subseteq V$ such that there does not exist an edge $e \in E$ with both endpoints in *S*.

INDEPENDENT SET PROBLEM:

Instance: Graph $G=(V,E)$, $k \in \mathbb{N}$

Question: Does *G* admit an independent set of size *k*?

a Prove that the INDEPENDENT SET PROBLEM $\in$ NP.

b INDEPENDENT SET CONSTRUCTION PROBLEM:

Input: Graph $G=(V,E)$

Output: A maximum independent set of *G*.

Prove that

INDEPENDENT SET CONSTRUCTION PROBLEM $\leq_P$ INDEPENDENT SET PROBLEM

CS524

Solutions to Final Exam

1. Do a depth first search and quit, **return**ing false as soon as a backedge is encountered (indicating the existence of a cycle). If no backedge is ever encountered, then **return** true. No more than $|V|-1$ edges can ever be examined before a cycle must be encountered.

2. Replace each vertex $v \in V$ with a pair of vertices, v_{in} and v_{out}, and an edge between v_{in} and v_{out} of weight $w(v)$. There are no costs associated with vertices in the new graph. Edges incident to v are now incident to v_{in} and v_{out}. The shortest path from σ to τ in the original graph corresponds to the shortest path from σ_{in} to τ_{out} in the new graph, and it has the same cost. We find this path using Dijkstra's Algorithm, which has a complexity in $O\left(|V|^2\right)$.

3. No, you should not use the FLOYD-WARSHALL ALGORITHM. It uses time in $\Theta\left(|V|^3\right)$. Using breadth first search to find the shortest path from any fixed vertex to all other vertices uses time in $O\left(|V|+|E|\right)=O\left(|V|\right)$, because G is sparse. Repeating this process $|V|$ times (once for each row of the distance matrix) requires time in $O\left(|V|^2\right)$.

4. ***a*** A certificate for G admitting an independent set of size k is an independent set S, $|S|=k$. The fact that no $e \in E$ has both endpoints in S can be verified in time in $O\left(|V|+|E|\right)$.

b Assume we have a program f to test if G admits an independent set of size k. We first determine K, the size of a largest independent set.

for $k \leftarrow 1$ **to** $|V|$ **do if** $f(G,k)$ **then** $\mathrm{K} \leftarrow k$

We then remove vertices from G, without descreasing the size of a maximum independent set, until only a maximal independent set remains. Let G-v denote the graph obtained by removing v, along with all incident edges, from V.

for each $v \in V$ **do if** $f(G-v,\mathrm{K}) = f(G,\mathrm{K})$ **then** $G \leftarrow G-v$

return $V(G)$ (return the vertices of G)