CS2223
FINAL EXAM
Name__________________

Date: December 16, 2004
All documentation permitted

1                 ________

2                 ________

3                 ________

4                 ________

TOTAL                 ________
1. (25 points) Describe an algorithm to test if a graph $G = (V, E)$ has a cycle. For example, for the graph

```
1.23
```

your algorithm should return `true`. The worst case execution time of your algorithm should be bounded by $O(|V| + |E|)$. 
2. (25 points) Prove or give a counterexample to the following.

**Conjecture:** For any weighted graph $G = (V, E)$ with $w: E \rightarrow \mathbb{R}^+$ such that the weights of all edges are distinct (that is, $e \neq e' \Rightarrow w(e) \neq w(e')$), and for any pair of vertices $v, w \in V$, all the edges of a shortest path from $u$ to $v$ must belong to a minimum spanning tree of $G$. 
3. (25 points) Suppose that you are given a directed graph \( G = (V, E) \), and you want to determine for each pair of vertices \( u, v \in V \) whether there is a path from \( u \) to \( v \). In particular, we want to compute entries for the Boolean array \( P : V \times V \rightarrow \{\text{true}, \text{false}\} \) such that for each \( u, v \in V \), \( P[u, v] = \text{true} \) if and only if there is a path from \( u \) to \( v \). For example, for the graph

\[
P[a, c] = \text{true} \quad \text{because of the path} \quad (a, b), (b, e), (e, c), \quad \text{but} \quad P[f, c] = \text{false} \quad \text{because there does not exist a path from} \quad f \quad \text{to} \quad c.
\]

Describe an algorithm to compute \( P \) with worst-case time complexity in \( O(|V|^3) \).
4. (25 points) For the Knapsack Problem, we are given positive integer \( n \) and \( n \) objects with positive weights \( w_1, \ldots, w_n \) and positive values \( v_1, \ldots, v_n \), as well as a knapsack with positive capacity \( W \). We seek a set of integers \( x_1, \ldots, x_n \) such that \( x_i \) is 0 or 1 for \( 1 \leq i \leq n \) to maximize

\[
\sum_{i=1}^{n} x_i v_i \quad \text{subject to} \quad \sum_{i=1}^{n} x_i w_i \leq W.
\]

Consider the following divide-and-conquer strategy to solve the Knapsack Problem (you may assume that \( W \) is a power of 2):

- If \( W = 1 \) and there is at least one object of weight 1, then add a most valuable object of weight 1 to the knapsack and remove it from the set of possible objects.
- If \( W > 1 \), then solve two Knapsack Problems with knapsacks of size \( W/2 \) and combine the two solutions.

Prove or give a counterexample to the following CONJECTURE: The above algorithm is guaranteed to give an optimal solution to any instance of the Knapsack Problem.
1. Do a depth first search of $G$, which has a cycle if and only if the depth first search forest has a backedge.

2. The CONJECTURE is false. The edge between $u$ and $v$ in the following graph, which is the shortest path from $u$ to $v$, does not belong to the minimum spanning tree.

3. Apply the Floyd-Warshall algorithm to the binary array with 0’s along the diagonal ($D[u,u]=0$ for all $u \in V$) and for $u \neq v$, $D[u,v]=1$ if $(u,v) \in E$ and $\infty$ otherwise. After applying the Floyd-Warshall algorithm, we obtain $P$ from $D$, by setting $P[u,v]=true$ if $D[u,v]$ is finite and false otherwise.

4. The algorithm rarely gives a correct solution. In fact, since it can only add objects of weight 1 it fails for the example $n=1$, $w_1=2$, $v_1=1$, $W=2$ since it will never add the object.