CSE 326 – Data Structures
Winter 2004
Dry assignment #5. Due date: 3/5/03

1. A weighted undirected graph G=(V,E) is given by the following adjacency matrix, M, in which M=(j,i)=M(i,j)=c(i,j). Use Dijkstra's algorithm to find the length of the shortest paths from S to any other vertex in G. For each vertex v, describe λ(v) at each stage of the algorithm, that is, after each execution of step 4 of the algorithm shown in class.
Remark: M(i,j)= ∞ means that (i,j)$\notin$E.

	S	A	B	C	D	E
S	∞	3	1	∞	8	∞
A	3	∞	∞	2	6	∞
B	1	∞	∞	5	∞	2
C	∞	2	5	∞	∞	4
D	8	6	∞	∞	∞	4
E	∞	∞	2	4	4	∞

2. Let $G=(V, E)$ be an undirected graph with non-negative weights on the edges.
Let $s,t \in V$, $e \in E$. Give efficient algorithms for each of the following problems:

- **a.** Does *e* belong to *all* shortest paths connecting *s* and *t*?
- **b.** Does *e* belong to *some* shortest path connecting *s* and *t*?

Explain why your algorithms are correct and analyze their time complexity.

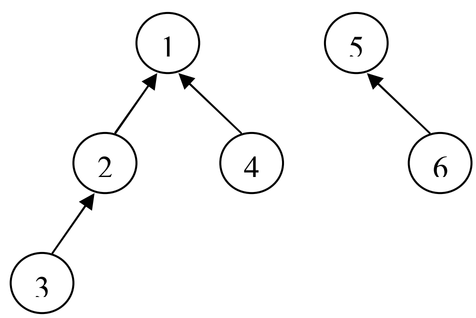
The more efficient your algorithms are, the more points you are going to receive.

3. How can we use the Bellman-Ford algorithm to detect in $O(|E|*|V|)$ steps if a given directed graph contains a negative cycle? Explain.

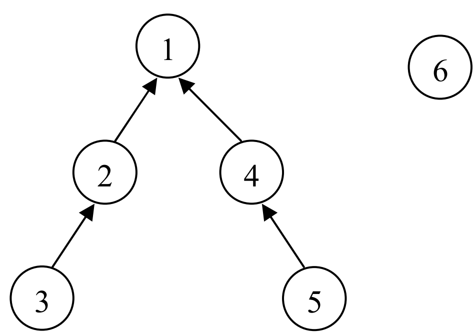
4. Consider an implementation of Disjoint Union/Find with trees and union-by-size (that is, in a union operation we make the smaller tree a subset of the larger). Initialy, there are six elements, each in a different set. {1},{2},{3},{4},{5},{6}.

For each of the following states, write a possible sequence of union operations that could create it, or explain why such a sequence does not exists.

I.



II.



III.