**9.2.1** **a: there are 9 arcs**
**b: 2 acyclic paths a-d, a-b-c-d**
**c: a,e**
**d: f,c**
**e: a-b-f-a, a-b-c-d-e-f-a, b-c-d-e-b, a-d-e-f-a**
**f: f-a-b-f-a-b-f (3 of this)(could be f-a-b-f)**

**9.3.5** **Prove:** $\sum_{i=0}^{n} def(V_i) == 2 \mid E_n \mid$ **for a graph G = (V, E) and n** $\geq 1$
**Base case:**
$V == \emptyset E == \emptyset$
$deg(V) == 2|D|$
$0 == 0$
**Assume:** $\sum_{i=0}^{k} deg(V_i) == 2|E_k|$ **for a graph G = (V, E) and n** $\geq 1$
**Prove:** $\sum_{i=0}^{k+1} deg(V_i) == 2|E_{k+1}|$ **for a graph G = (V, E) and n** $\geq 1$
$\sum_{i=0}^{k} deg(V_i) + deg(V_{k+1})$ **by definition of sum**
$2|E_k| + deg(V_{k+1})$ **by inductive hypothesis**
**Here we must ask the question: What exactly does** $deg(V_{k+1})$ **equal? We know that for and undirected graph if** $\exists E(u, v)$ **then** $\exists E(v, u)$. **This is also true for a vertex pointing to itself (it gets counted twice). We also know that the edge set** $E_k$ **can't get smaller with the addition of vertex** $V_{k+1}$. **This means the addition of any edges by vertex** $V_{k+1}$ **may grow the set or may not grow the set. With all of this being said,** $deg(V_{k+1})$ **could equal to 0 or 2 or 4 ... or 2k verticies. In any case, this MUST be the new value of** $2|E_{k+1}|$
$2|E_k| + deg(V_{k+1}) == 2|E_{k+1}|$

**9.4.1** **Constructed Graph:**
**Marquette**
**I I Excabana**
**I Sault Ste. Marie**
**Menominee**
**.**
**Grand Rapids**
**I I I I I Battle Creek**
**I I I I Detroit**
**I I I Lansing**
**I I Ann Arbor**
**I Kalamazoo**
**Saginov**
**I**
**Flint**

**1.3** If my answer is TRUE, I must show that: for every shorest path tree, this case must hold

If my answer is FALSE, I must show that: a counter example exists, namely there exists at least one representation of an adjacency list that does not yield a shortest path tree run from s.

The answer is TRUE

Starting small: suppose a graph with no edges. The adjacency list genereated for any node would be empty. Now, suppose a graph with an arbitrary amount of edges, but none include the source vertex. again, the adjacency list is empty because the source vertex does not point to any node nor does it point to itself

Finally, suppose that s points to an arbitrary positive non-zero amount of nodes. Based on the BFS algorithm, the neighbors of s will populate the shortest path's tree. Whether or not this tree finds itself to the desired node is beside the point. Since BFS CAN use an adjacency list representation, the realistic success of this method is dependent on the programmer.

**2.1 A. TRUE, for a finite graph.**

**suppose a graph with V of finite size n. first, assume that every node points to at least one other node (the (out)deg(ALL VERTICIES) != 0). take a vertex out of V. It can't point to itself, so it has n-1 choices to point to. In this example, there are n verticies and at least n edges. The edge set contains an edge** $(u, v)$ **where** $u \neq v$**. As the edges are enumerated, the last edge (one that doesn't have an outdegree), will have have an empty verticie set to point to and this resulting edge must create a cycle. If the last vertex did not point to anything (it has an (out)deg == 0, the graph remains a DAG and we are left with one vertex, u, where** $deg(u), == 0$

**2.2 B. FALSE**

**suppose the graph G with set V = a,b,c,d and E = (a,b)(b,d)(d,a)**
**c has an (out)deg of zero and the graph is cyclic from a-b-d-a-d-b-...**
**therefore, this statement is false.**

**2.3 C. TRUE**

**see questions 1.C. the arguement shows that, in the addition of vertex, there will always be a last "vertex" that must point to another node which would cause a cycle. If all verticies belong to some outgoing edge, there are n edges but n-1 nodes to point to. 1 edge remains and this edge results in a cycle.**