Homework 7 Theory

Joseph Mulray

Febuary 28, 2017

Problem 5.20:

> Write a program using the MFSET operations that computes the sets of equivalent states of a given finite automaton.

```
def equivalent_states(inputs, transitions):
    for p in inputs:
        for q in inputs:
        #if they are the same state
            if p == q
                #check if both non accepting or accepting
                if transition(p) != transition(q):
                    #not equivalent merge both of them
                    MERGE(p,q)
                else:
                    pass
            else:
                #different states, merge both
                MERGE(p,q)
```

Problem 2:

> Consider an undirected graph G = (V,E) with n = IVI and m = IEI. The degree of a vertex is the number of edges incident on that vertex. Let di be the degree of vertex vi, Show that $SUM[1..n](di) = 2m$

An undirected edge that connects to two vertices together has a degree of 2. The degree its 2 because each vertex is incident on that vertex summing up to 2, 1 + 1, for each vertex. For each edge, m, there is 2m that it each edge contributes to. Therefore $SUM[1..n](di) = 2m(edges)$

Problem 3:

> In a directed graph, we can talk about in-degree and out-degree, the number of edges, respectively, arriving and leaving a given vertex. Show that the sum of the in-degrees of a graph is equal to the sum of the out-degrees.

One out-degree edge has an in-degree for one vertex and an out-degree for another vertex. This means that one out-degree edge in a directed graph will have $SUM[1..n]$ where n is the number of out-degree edges will equal to m edges.

Problem 6.1:

> By an adjacency matrix giving arc costs:

	a	**b**	**c**	**d**	**e**	**f**
a		3		4		5
b			1			1
c				2		
d		3				
e				3		2
f				2		

> By a linked adjacency list with arc costs indicated:

a	=> (a,0) => (b,3) => (d,4) => (f,5)
b	=> (b,0) => (c,1) => (f,1)
c	=> (c,0) => (d,2)
d	=> (b,3) => (d,0)
e	=> (d,3) => (e,0) => (f,2)
f	=> (d,2) => (f,0)