**Homework 2 Morally Due Feb 12**

1. (20 points) The alphabet is $\{0, \ldots, 9\}$. We interpret the input as a base 10 natural number, read *right to left*. So the number 29139 will be read 9-3-1-9-2.

   Give the diagram for a finite automata classifier that determines, given $w$, what $w$ is congruent to mod 6. How many states does it have?

   **SOLUTION TO PROBLEM 1**

   $10^0 \equiv 1 \pmod 6$

   $10^1 \equiv 4 \pmod 6$

   $10^2 \equiv 4 \pmod 6$

   etc.

   So we will keep a running sum of $d_0 + 4d_1 + 4d_2 + \cdots \pmod 6$.

   We OMIT the rest.

2. (30 points) We have seen some number $m$ such that the DECIDER (of minimal number of states) for Mod $m$ (is $x \equiv 0 \pmod m$) has LESS states than the CLASSIFIER (of minimal number of states) for mod $m$.

   (a) (15 points) Give an infinite number of examples of $m$, with $m \geq 100$, where the Decider will have LESS states than the classifier. Explain why this is, but you do not need to draw the DFA's.

   (b) (15 points) Give an infinite number of examples of $m$, with $m \geq 100$, where the Decider will have AS MANY states as the classifier. Explain why this is, but you do not need to draw the DFA's.

   (c) (0 points) Think about: Fill in the following statement:
   *The DECIDER for Mod m will have LESS states than the CLASSIFIER iff XXX.*

   **SOLUTION TO PROBLEM 2**

   a) LESS: The key is that if the powers $10^i \pmod m$ are eventually all 0's then you no longer need to keep track of the weighted sum and can just go to a 0 or non-zero state.

Examples:

$m = 100$: $10^0 \equiv 1$, $10^1 \equiv 10$, $10^2 \equiv 0$, and for all $i \geq 2$, $10^i \equiv 0$. So the DFA will just check if the $d_0$ and $d_1$ digits are 0. If they are then goto a state ACCEPT. If not then goto REJECT. All later transitions are self loops.

$m = 1000$ and $m = 10000$ and $m = 10^i$ are similar.

b) THE SAME: $m$ any prime $> 100$ will work. There are other values that work as well. Justification omitted.

c) XXX is

$\gcd(m, 10) > 1$ *and* $m > 2$. Justification omitted.

3. (25 points) Prove using NFA's: If $L$ is regular then $L^*$ is regular. Begin with an NFA for $L$ and then modify it to get an NFA for $L^*$. Formally write the new NFA in terms of the old one. Draw a picture also.

**SOLUTION TO PROBLEM 3**

The NFA for $L$ is $(Q, \Sigma, \Delta, s, F)$. There are two things you need to do:

- We want the empty string accepted. Hence create a NEW start state, called NEW, that is also a final state. Add an e-transition from it to the original start state.
- Add an e-transition from all final states to the NEW start state. (The old one would also have been fine.)

Formally the new NFA is

$(Q \cup \{NEW\}, \Sigma, \Delta', NEW, F \cup \{NEW\})$.

Where

$\Delta'(q, \sigma) = \Delta(q, \sigma)$ for all $q \in Q$

$\Delta'(q, e) = \Delta(q, e) \cup \{NEW\}$ for all $q \in F$

$\Delta'(NEW, e) = \{s\}$.

4. (25 points) A JUSTIN-NFA is an NFA that has no e-transitions. Give a procedure that takes an NFA and produces an equivalent JUSTIN-NFA. (Note- be careful. A sequence of e-transitions can go through many states.)

**SOLUTION TO PROBLEM FOUR**

Let $L$ be accpeted by NFA $(Q, \Sigma, \Delta, s, F)$.

The equivalent JUSTIN-NFA is $(2^Q, \Sigma, \Delta', \{s\}, F')$. where

$\Delta'(A, \sigma)$ is the set of ALL states you can get to from any $q \in A$ using any string of the form $e^i \sigma e^j$ in the old NFA.

$F'$ is the set of all $A$ such that $A \cup F \neq \emptyset$.

**END OF SOLUTION TO PROBLEM FOUR**