MA/CSSE 474 Day 04 Summary

1. Show that $\equiv_3$ is an equivalence relation. ($a \equiv_3 b$ iff $b-a = 3k$ for some integer k).
   reflexive:

   symmetric:

   transitive:

2. Can a language be uncountable? Is the set of languages over a specific alphabet uncountable?

3. Questions about **maxstring** function:
   maxstring(A^nB^n) = *maxstring*({a}*) = (later, to check your understanding): *maxstring*($\{b^na: n\geq 0\}$) =
   Let INF be the set of infinite languages. Let FIN be the set of finite languages.
   Are the language classes FIN and INF closed under *maxstring*?

4. Questions about **chop** function:
   What is *chop*(A^nB^n)? What is *chop*($A^nB^nC^n$)?
   Are FIN and INF closed under *chop*?

5. Questions about **firstchars** function:
   What is *firstchars*(A^nB^n)? What is *firstchars*({a, b}*)?
   Are FIN and INF closed under *firstchars*?

6. A decision problem is a problem whose answer is _____________.

7. What does <x> mean? <x, y>?

8. Some decision problems:
   a. Consecutive pair of factors
   b. Halting problem
   c. Web pattern matching
   d. Primality testing
   e. Graph connectivity
   f. Multiplication as decision problem
   g. Sorting as decision problem

9. Constructing one machine based on another machine

Consider the multiplication language:

INTEGERPROD = {*w* of the form <$integer_1$>x<$integer_2$>=<$integer_3$>, where:
<$integer_n$> is any well-formed integer representation and $integer_3 = integer_1 * integer_2$}

Given a multiplication procedure, we can build a language recognition procedure?

Given the language recognition procedure, we can build a multiplication procedure:

10. A configuration of a DFSM M is an element of $K \times \Sigma^*$. Contains all info needed to complete the computation. Initial configuration of M: (s_M, w), Where s_M is the start state of M.

Recap - Definition of a DFSM

$M = (K, \Sigma, \delta, s, A)$, where:

The D is for Deterministic

K is a finite set of ***states***

Σ is a (finite) ***alphabet***

$s \in K$ is the ***initial state*** (a.k.a. start state)

$A \subseteq K$ is the set of ***accepting states***

$\delta: (K \times \Sigma) \rightarrow K$ is the ***transition function***

Sometimes we will put an M subscript on K, Σ, δ, s, o
A (for example, s_M), to indicate that this component i
part of machine M.

11. The *yields-in-one-step* relation: $|\text{-}_M$:
(q, w) $|\text{-}_M$ (q', w') iff
   - $w = a\,w'$ for some symbol $a \in \Sigma$, and
   - $\delta(q, a) = q'$
12. The *yields-in-zero-or-more-steps* relation: $|\text{-}_M{}^*$
$|\text{-}_M{}^*$ is the reflexive, transitive closure of $|\text{-}_M$.

13. A ***computation*** by *M* is a finite sequence of configurations $C_0, C_1, \ldots, C_n$ for some $n \geq 0$ such that:
   - C_0 is an initial configuration,
   - C_n is of the form (q, ε), for some state $q \in K_M$,
   - $\forall i \in \{0, 1, \ldots, n\text{-}1\}$ $(C_i$ $|\text{-}_M$ $C_{i+1})$

14. A DFSM *M* ***accepts*** a string *w* iff (s_M, w) $|\text{-}_M{}^*$ (q, ε), for some $q \in A_M$
rejects *w* iff (s_M, w) $|\text{-}_M{}^*$ (q, ε), for some $q \notin A_M$. The ***language accepted by*** *M*, denoted $L(M)$, is the set of all strings accepted by *M*. A language is ***regular*** if it is L(M) for dome DFSM M.

15. ***Theorem:*** Every DFSM *M*, in configuration (q, w), halts after $|w|$ steps.

16. $L = \{w \in \{0, 1\}^* : w$ has odd parity$\}$. I.e. an odd number of 1's.

17. $L = \{w \in \{a, b\}^* :$ no two consecutive characters are the same$\}$.

18. $L = \{w \in \{a, b\}^* : \#_a(w) >= \#_b(w)\}$.

19. $L = \{w \in \{a, b\}^* : \forall x,y \in \{a, b\}^* (w=xy \rightarrow |\#_a(w) - \#_b(w)| <= 2)\}$