Solution sketch 2 - Computational Models - Spring 2017

1. (a) L_1 is not regular. Can be proved by Myhill-Nerode Theorem. Note that for every $m_1 \neq m_2$, $b^{m_1} \not\sim_{L_1} b^{m_2}$.

   (b) L_2 is not regular. Can be proved by the pumping lemma. For any $l \geq 0$ choose $s = 0^l110^l$. Let $s = xyz$ s.t. $|y| > 0$ $|xy| \leq l$. Choose $i = 2$. $xy^iz \notin L$ (prove it).

   (c) L_3 is regular. Trivial.

   (d) L_4 is not regular. Use the pumping lemma. Almost similar to the proof for the same language with n^2 (done in the lecture).

   (e) L_5 is not regular. Can be proved by the pumping lemma. For any $l \geq 0$ choose $s = 0^l1^l$. Let $s = xyz$ s.t. $|y| > 0$ $|xy| \leq l$. Choose $i = 0$. $xy^iz \notin L$ (prove it).

   (f) L_6 is not regular. Can be proved by Myhill-Nerode Theorem. For every $i \neq j$, $[1^i]$ is not equivalent to $[1^j]$ by look at $z = 0 + 1^i$.

   (g) L_7 is not regular. Can be proved using the homomorphism $h : \Sigma \to \{0,1\}^*$. $h(a) = 0, h(b) = 0, h(c) = 1, h(d) = 1$ and that $L = \{0^n1^n \mid n \leq 0\}$ isn't regular.

2. (a) i. $(bb)^*$

      ii. $(01110111 \cup 11011101)$

   (b) $(ba)^*$. Steps of proof:

      i. Prove by induction of word length that:

      $$L = \{w \ : \ \textit{any maximum sequence of consecutive 0's in w, has an even length}\}$$

      ii. Prove that $L((ba)^*) \subseteq h^{-1}(L)$ (for every $w \in ((ba)^*), h(w) \in L$).

      iii. Prove by induction of word length that $h^{-1}(L) \subseteq L((ba)^*)$. Use 2(b)i in the induction step.

3. $\{w : |w| \; mod \; 3 = 0\}$, $\{w : |w| \; mod \; 3 = 1\}$, $\{w : |w| \; mod \; 3 = 2\}$.
   prove that $w_1 \sim^L w_2$ iff w_1 and w_2 are in the same group.

4. (a) The regular languages are closed under this operation. Let $A = \langle Q, \Sigma, \delta, q_0, F \rangle$ be a DFA for L. Assume $Q = \{q_0, ..., q_n\}$. We build an NFA, A' for $Inv(L)$ as follows:
   - A' has 3 parts:
     - Part 1: A copy of A with no accepting states.
     - Part 2: For every $1 \leq i, j \leq n$ we will have an automata B_{ij} which is a copy of an automata for $Reverse(L)$ (how?).
     - Part 3: A copy of A.
   - For every $1 \leq i, j \leq n$ we will have an ϵ transition from q_i in part 1 to q_j in B_{ij}, and an ϵ transition from q_i in B_{ij} to q_j in part 3.

   Prove that indeed $L(A') = Inv(L)$. Understand why does this gives an NFA for $Inv(L)$.

   (b) True. Take the DFA for L from Myhill-Nerode theorem and change his accepting states to be the states that correlates to C_i for each $i \in I$.

   (c) False. $L = \{0^n 1^n : n \geq 0\}$ is non regular over $\Sigma = \{0, 1\}$, but for $h(0) = 1$, $h(1) = 1$, $h(L)$ is regular.

5. (a) Correct. let $M = (Q, \Sigma, \delta, q_0, F)$ the DFA with the minimal number of states that accept L. from Myhill-Nerode theorem $rank(L) = |Q|$. By replacing accepted with unaccepted states of M we get a DFA for $\overline{L}$ with $|Q|$ states. Thus the minimal number of states for DFA that accepts $\overline{L}$ is no more then $rank(L)$. This is why $rank(L) \leq rank(\overline{L})$. From the same claim for $\overline{L}$ we get $rank(\overline{L}) \leq rank(\overline{\overline{L}})$ and finally $rank(\overline{L}) = rank(L)$.

   (b) Correct. from Myhill-Nerode theorem exists M_1 and M_2 that accepts L_1 and L_2 such that $|Q_i| = rank(L_i)$. As seen in class we can build a product DFA from M_1 and M_2 with $|Q_1| * |Q_2|$ states that accept $L_1 \cap L_2$. Thus, in the minimal DFA for $L_1 \cap L_2$ there is no more then $|Q_1| * |Q_2|$ states. By Myhill-Nerode theorem $rank(L_1 \cap L_2) \leq rank(L_1) \cdot rank(L_2)$.

   (c) False. For the language $L = \{0\}$, the DFA with the minimal number of states that accepts L have 3 states. Thus, by Myhill-Nerode theorem $rank(L) = 3$. But, there is a NFA with 2 states that accepts L.

6. (a) If $\exists w. |w| > n$ then as we learned in the pumping lemma, this word can be pumped infinitely and therefore $L(A)$ is infinite. if $L(A)$ is infinite, then there is a word $w \in L(A)$ such that $|w| > n$. The run of this word

in A contains a cycle. We remove all cycles from the run and remember one simple cycle c, $|c| \leq n$. The run without the cycles give a word $w' \in L(A), |w'| < n$. We start pumping w' with the cycle c and we will eventually get a word in $L(A)$ in the proper length.

(b) Given a DFA A, we can run in A all the words w, such that $n < |w| \leq 2n$. If one of the words is accepted, then $L(A)$ is infinite, otherwise - finite.

(c) First we check if $L(A)$ is infinite. If it is, we return $false$. otherwise, we run in A all words of length at most n and count how many are accepted. we return $true$ iff the count is 9,122,009.

(d) Note that $L(A_1) = L(A_2) \leftrightarrow L(A_1) \subseteq L(A_2) \wedge L(A_2) \subseteq L(A_1) \leftrightarrow L(A_1) \backslash L(A_2) = \emptyset \wedge L(A_2) \backslash L(A_1) = \emptyset \leftrightarrow (L(A_1) \backslash L(A_2)) \cup (L(A_2) \backslash L(A_1)) = \emptyset$. To check if $L(A1) = L(A2)$ construct the DFA D for $(L(A1) \backslash L(A2)) \cup (L(A2) \backslash L(A1))$ and check if $L(D)$ is empty (how?).