COMP 280 : Assignment 12 - Sample Solutions

1. The language $\{a^n ba\}$ is the regular language a^*ba, so there exists a (simple three state) FA that accepts it.

2. The set of all palindromes over Σ of length at most three is finite, there is therefore a FA that accepts it.

3. To accept a palindrome like $a^n ba^n$ for all n, a FA would have to count the number of a's in the prefix, and compare it to the number of a's in the suffix. This means that for any n, the FA has to be able to store the value of n, which implies an infinite number of states. So no FA can accept the language of all palindromes.

4. Since the FAs have N states and are defined over an alphabet Σ with S symbols, the transition table for one of these FAs will have NS (possibly empty) entries.

   There are also N possible choices for the start state, and 2^N possible combinations of final states (since each of the N states may or may not be a final state).

   (a) If the table can have empty entries and multiple transitions per entry, then for each entry in the table we have 2^N possible combinations (since it may or may not have a transition to any of the N states). Since there are NS entries, we get $(2^N)^{NS}$ possible transition tables, so there are $N2^{N+N^2S}$ possible FAs.

   (b) If the table has to have exactly one transition per entry, then there are for each entry N possibilities. So we have N^{NS} possible tables, so $N^{NS+1}2^N$ different FAs.

5. Let $M_1 = (Q_1, \Sigma, \delta_1, q_1, F_1)$ and $M_2 = (Q_2, \Sigma, \delta_2, q_2, F_2)$. Then M is defined as $(Q_1 \times Q_2, \Sigma, \delta, [q_1, q_2], F_1 \times F_2)$, with $\forall p_1 \in Q_1\ \forall p_2 \in Q_2\ \forall a \in \Sigma : \delta([p_1, p_2], a) = [\delta_1(p_1, a), \delta_2(p_2, a)]$.

   For some input sequence I, if M accepts I, then there has to be a sequence $\{[q_1, q_2], \ldots, [f_1, f_2]\}$ (with $[f_1, f_2] \in F_1 \times F_2$) of states of M which is traversed on input I. Then, by definition of δ, M_1 will go through the sequence of states $\{q_1, \ldots, f_1\}$ on input I. Since $f_1 \in F_1$, M_1 will accept I.

6. M1 and M2 both accept the input sequence "aa", but M3 does not. So M3 does not accept the same language as M1 or M2.

   If you number the states of M1 and M2 from left to right and top to bottom, then: if you merge state 2 and 3 of M2 into state 2', state 5 and 6 of M2 into state 4', then merge state 4 and 4' into 4", you will get M1. So M1 and M2 accept the same language (a more mathematical way to prove this is by minimizing M1 and M2, which will do the above merges automatically).