Solutions to Homework 2

Problem 1.

a. A TM M_1 to decide $\{w|w$ contains twice as many 0s as 1s$\}$ operates as follows:

On input w,

1. Scan right until a 1 is encountered, and mark it with a special character. If we reach the end of the input before reading a 1, go to step 3.
2. Move the tape head to the left end of the tape. Scan right, attempting to mark two 0 characters. If successful, reset tape head and return to step 1.
3. Reset tape head and make sure all characters are marked. If so, accept. If not, reject.

b. A TM M_2 to decide $\{w|w$ does not contain twice as many 0s as 1s$\}$ may simply run M_1 on its input; if M_1 accepts, M_2 rejects, and if M_1 rejects, M_2 accepts.

Problem 2.

Turing machines with stay put instead of left (TMSPs) can only recognize regular languages. For every DFA D, there clearly exists a TMSP T that accepts the same language, since a DFA is simply a Turing machine with a read-only tape and a tape head which only moves to the right.

We now demonstrate that for every TSMP $T = (Q, \Sigma, \Gamma, \delta, q_0, q_{accept}, q_{reject})$, there exists an NFA $N = (Q + \{q_{rej}\}, \Gamma, \delta_n, q_0, F_n)$ (and hence a DFA D) that accepts the same language. We define a *maximal chain* of stay-put transitions to be a chain of transitions $\delta(q_{i_1}, \alpha_1) = (q_{i_2}, \beta_1, S), \delta(q_{i_2}, \alpha_2) = (q_{i_3}, \beta_2, S), \ldots \delta(q_{i_n}, \alpha_n) = (q_{i_{n+1}}, \beta_n, R)$ such that no transition is repeated, and n is the largest possible value for a chain ending with that particular final transition. We may construct N as follows:

1. If $\delta(q_i, \alpha) = (q_j, \beta, R)$ then $q_j \in \delta_n(q_i, \alpha)$, since β is never again seen by T.
2. For any maximal chain of stay-put transitions starting with state q_i and character α, and ending with state q_j, place $q_j \in \delta_n(q_i, \alpha)$. Since only move-right transitions (for which T will consume an input character it did not produce) affect the overall computation, this "chain-skipping" will keep $L(N) = L(T)$.
3. $q_{accept} \in F_n$
4. For every state of N not having an outgoing transition for some character $\gamma \in \Gamma$, add a transition to the special q_{rej} state. (Recall that a TM may halt before examining all of its input, while an FSM must read all input.) q_{rej} has transitions to itself for all characters in Γ.

Problem 3.

Suppose we have a machine E which enumerators the strings in some language L in lexicographic order. We can then build a TM D to decide L. On input x, D runs E. After each string w output by E, D tests if $x = w$. If so, D halts and accepts. If w is lexicographically less than x, D repeats. If not, D halts and rejects. If E runs out of strings, D halts and rejects. Since E will eventually either run out of strings or output a string lexicographically greater than x, D will halt on all strings, accepting only those in L.

Suppose, on the other hand, that we have a TM D that decides L. Let s_i denote the i^{th} lexicographically ordered string in Σ^*. We can construct a machine E to enumerate the strings of L in lexicographic order. E simply runs D on the strings s_i in order, waiting for D to halt before proceeding to the next string in Σ^*. If D accepts a particular s_i, E outputs it. Since D is a decider, D is guaranteed to halt on all inputs; since E submits strings to D in lexicographic order, E will clearly output all strings in L in lexicographic order.

Problem 4.

Turing machines that cannot write on the input portion of their tapes (TMNWs) can only recognize regular language. For every DFA D, there clearly exists a TMNW T that accepts the same language, since a DFA is simply a Turing machine with a read-only tape and tape head which can only move right.

We now demonstrate that for every TMNW T there exists an NFA N (and hence a DFA D) that accepts the same language. To do so, we first consider a more limited version of a Turing machine which can only read from its tape, but cannot write to it anywhere. (We call this a TMRO. If you have a problem with our name for it, don't complain. It will only make things worse.) Clearly, for every DFA D there exists a TMRO T that accepts the same language.

Problem 5.

Regardless of whether or not the universe expands forever, the language A contains only one string. Since any finite language is decidable (in fact, is regular), A is a decidable language.