Problem 1

Give the sequence of configurations that TM M_2 (EX 3.4) enters when started on the indicated input string.

a) 0

$q_1 0$, $\sqcup q_2 \sqcup$, $\sqcup \sqcup q_{accept} \sqcup$

b) 00

$q_1 00$, $\sqcup q_2 0$, $\sqcup x q_3 \sqcup$, $\sqcup q_5 x \sqcup$, $q_5 \sqcup x \sqcup$, $\sqcup q_2 x \sqcup$
$\sqcup x q_2 \sqcup$, $\sqcup x \sqcup q_{accept} \sqcup$.

c) 000

$q_1 000$, $\sqcup q_2 00$, $\sqcup x q_3 0$, $\sqcup x 0 q_4 \sqcup$, $\sqcup x 0 \sqcup q_{reject} \sqcup$

d) 000000

$q_1 000000$, $\sqcup q_2 00000$, $\sqcup x q_3 0000$, $\sqcup x 0 q_4 000$, $\sqcup x 0 x q_3 00$
$\sqcup x 0 x 0 q_4 0$, $\sqcup x 0 x 0 x q_3 \sqcup$, $\sqcup x 0 x 0 q_5 x \sqcup$,
$\sqcup x 0 x q_5 0 x \sqcup$, $\sqcup x 0 q_5 x 0 x \sqcup$, $\sqcup x q_5 0 x 0 x \sqcup$, $\sqcup q_5 x 0 x 0 x \sqcup$,
$q_5 \sqcup x 0 x 0 x \sqcup$, $\sqcup q_2 x 0 x 0 x \sqcup$, $\sqcup x q_2 0 x 0 x \sqcup$, $\sqcup x x q_3 x 0 x \sqcup$
$\sqcup x x x q_3 0 x \sqcup$, $\sqcup x x x 0 q_4 x \sqcup$, $\sqcup x x x 0 x q_4 \sqcup$, $\sqcup x x x 0 x \sqcup q_{reject} \sqcup$

Problem 2.

Give the sequence of configurations for TM M_1 (Ex 3.5) enters when started on the indicated input string.

a) 11

$q_1 11$, $\sqcup q_3 1$, $\sqcup 1 q_3 \sqcup$, $\sqcup 1 \sqcup q_{reject} \sqcup$

b) 1#1

$q_1 1\#1$, $\sqcup q_3 \#1$, $\sqcup \# q_5 1$, $\sqcup \# 1 q_5 \sqcup$, $\sqcup \# q_7 1 \sqcup$
$\sqcup q_7 \# 1 \sqcup$, $q_7 \sqcup \# 1 \sqcup$, $\sqcup q_9 \# 1 \sqcup$, $\sqcup \# q_{11} 1 \sqcup$, $\sqcup q_{12} \# x \sqcup$,
$q_{12} \sqcup \# x \sqcup$, $\sqcup q_{13} \# x \sqcup$, $\sqcup \# q_{14} x \sqcup$, $\sqcup \# x q_{14} \sqcup$, $\sqcup \# x \sqcup q_{accept} \sqcup$

c) 1##1

$q_1 1\#\#1$, $\sqcup q_3 \#\#1$, $\sqcup \# q_5 \# 1$, $\sqcup \#\# q_{reject} 1$

d) 10#11

$q_1 10\#11$, $\sqcup q_3 0\#11$, $\sqcup 0 q_3 \#11$, $\sqcup 0\# q_5 11$, $\sqcup 0\#1 q_5 1$,
$\sqcup 0\#11 q_5 \sqcup$, $\sqcup 0\# 1 q_7 1 \sqcup$, $\sqcup 0 \# q_7 11\sqcup$, $\sqcup 0 q_7 \#11\sqcup$,
$\sqcup q_7 0\#11\sqcup$, $q_7 \sqcup 0\#11\sqcup$, $\sqcup q_9 0\#11\sqcup$, $\sqcup 0 q_9 \#11\sqcup$
$\sqcup 0\# q_{11} 11\sqcup$, $\sqcup 0 q_{12} \#x1\sqcup$, $\sqcup q_{12} 0\#x1\sqcup$, $\sqcup q_{12} 0\#x1\sqcup$,
$q_{12}\sqcup 0\#x1\sqcup$, $\sqcup q_{13} 0\#x1\sqcup$, $\sqcup x q_8 \#x1\sqcup$, $\sqcup x\# q_{10} x1\sqcup$,
$\sqcup x\#x q_{10} 1\sqcup$, $\sqcup x\#x1 q_{reject}\sqcup$

e) Similar to above with last '1' replaced by '0' (10#10)
we get to $\sqcup x\#x q_{10} 0\sqcup$, $\sqcup x\# q_{12} xx\sqcup$, $\sqcup x q_{12} \#xx\sqcup$,
$\sqcup q_{12} x\#xx\sqcup$, $q_{12}\sqcup x\#xx\sqcup$, $\sqcup q_{13} x\#xx\sqcup$, $\sqcup x q_{13}\#xx\sqcup$,
$\sqcup x\# q_{14} xx\sqcup$, $\sqcup x\#x q_{14} x\sqcup$, $\sqcup x\#xx q_{14}\sqcup$, $\sqcup x\#xx\sqcup q_{accept}\sqcup$

Problem 3

Modify the proof of Thm 3.10 to obtain corollary 3.12 showing that a language is decidable iff some nondeterministic TM decides it.

In the proof of theorem 3.10, the constructed TM D does not halt, even if TM N halts on every branch. In fact D keeps trying all strings on tape 3 lexicographically.

If N halts on all branches, each branch will have finitely many steps. In addition, since at each point in time, only finitely many choices are available to N, the total number of steps in N's computation tree is finite. This means there exists an l such that every string of choices of length $> l$ is not needed (N reaches accept or reject after making at most l choices. We can modify D to try all strings of length $\leq l$ lexicographically on tape 3. This guarantees that D will stop. If D stops before accepting, it rejects.

Problem 5

a) Can a TM ever write the blanc symbol $\sqcup$ on its tape?

YES

b) Can the tape alphabet Γ be the same as the input alphabet Σ?

No. $\sqcup \in \Gamma$ and $\sqcup \notin \Sigma$.

c) Can a Turing machine head be in the same location in two successive steps?

YES. If it moves left on the left most square.

d) Can a Turing Machine contain just a single state?

No. At least we have q_{accept} & q_{reject}

Problem 6.

E = "Ignore the input.

1. Repeat the following for $i = 1, 2, 3, \ldots$
2. Run M on s_i
3. If it accepts, print out s_i."

If M recognizes the language, why isn't E an enumerator for it?

Because M might never halt on some input s_i and, therefore, $s_{i+1}, s_{i+2}, \ldots$ will never be considered.

Problem 7.

Why is the following not a ~~[illegible]~~ description of a legitimate TM.

M_{bad} = "The input is a polynomial p over $x_1, \ldots x_n$.

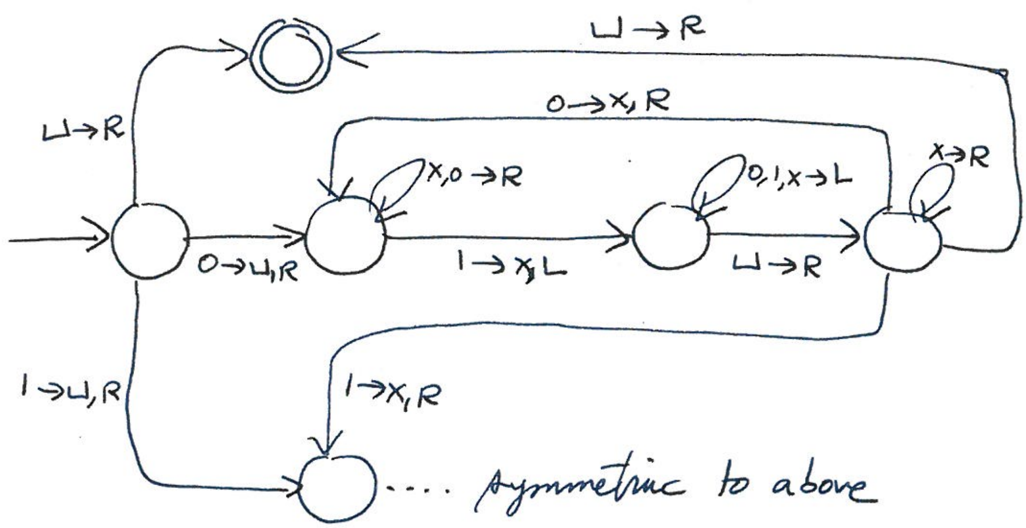
1. Try all possible settings of $x_1, \ldots x_n$ to integer values
2. Evaluate p on all these settings
3. If any of these settings evaluate to 0, accept; otherwise reject."

This description does not say how all the possible settings can be tried. A naive implementation might fail to reach a particular setting because there are infinitely many. One possibility is to say: ~~[illegible]~~

for every $i = 0, 1, 2, 3, \ldots$
try all settings for $x_1, \ldots x_n$ for integer values in $\{-i, \ldots, i\}$

Problem 8

a) $\{w \mid w$ contains equal number of 0s and 1s$\}$



Find the first unmarked bit. If it's 0, mark it and look for a 1. Mark the 1, and go back all the way to the left. Repeat. Transitions not shown go to q_{reject}.

b) $\{w \mid w$ contains twice as many 0s as 1s$\}$

Similar to above, but find first 2 unmarked bits. If 00, look for a 1 to mark. If 01 or 10, look for a 0 to mark. If 11 look for 4 zeros to mark.

c) Switch reject and accept states.