3.7

All possible settings of x_i are infinite, as there are infinite integers. Therefore, step 2 can never yield to step 3 in cases of reject.
However, in step 3, this machine needs to run for all cases of x_i to return reject. Therefore, it's not a legitimate Turing machine.

3.8b

1. Scan the tape and mark the first 1 that hasn't been marked. If no unmarked 1's are found, go to stage 5. Else move head back to start of tape.

2. Scan the tape until an unmarked 0 is found and mark it. If no 0s are found then **reject**.

3. Scan the tape once more until an unmarked 0 is found and mark it. If no 0s are found then **reject**.

4. Move the head back to the start of the tape and go to stage 1.

5. Move the head back to the start of the tape. Scan tape for unmarked 0s. If none are found then **accept**. Else **reject**.

3.8c

Run the Machine above and change accept to reject and via versa.

1. Scan the tape and mark the first 1 that hasn't been marked. If no unmarked 1's are found, go to stage 5. Else move head back to start of tape.

2. Scan the tape until an unmarked 0 is found and mark it. If no 0s are found then **accept**.

3. Scan the tape once more until an unmarked 0 is found and mark it. If no 0s are found then **accept**.

4. Move the head back to the start of the tape and go to stage 1.

5. Move the head back to the start of the tape. Scan tape for unmarked 0s. If none are found then **reject**. Else **accept**.

4.2

Let's define:
C = { $\langle M, R\rangle | M$ is a DFA and R is a RE with $L(M) = L(R)$ }.
Recall EQ_{DFA} = { $\langle A, B\rangle|$ A and B are DFAs and $L(A) = L(B)$ }. Then the following Turing machine T decides C:
On input $\langle M, R\rangle$, where M is a DFA and R is a regular expression:

1. Convert R into a DFA D_R.
2. Return $\langle M, D_R\rangle \in^? EQ_{DFA}$

4.3

M: On input $\langle A\rangle$, where A is a DFA and $L(A) = \sum^*$

1. Build B is DFA recognizes $\sum^*$
2. Return $\langle A, B\rangle \in^? EQ_{DFA}$

4.4

M: On input $\langle G\rangle$, where G is a CFG

1. Convert G into an equivalent CFG G' in Chomsky normal form.
2. If G' include rule $S \longrightarrow \epsilon$ where S is start symbol, accept. Else reject.

4.16

A: On input $\langle R\rangle$:

1. Construct $R = (0 \cup 1)^*111(0 \cup 1)^*$
2. Return $\langle R, T\rangle \in^? EQ_{RE}$

4.21

S: On input $\langle M\rangle$:

1. Construct DFA A that that accepts ω^R whenever it accepts ω
2. Return $\langle M, A\rangle \in^? EQ_{DFA}$