1. Problem 5.26

(a) If $M$ is a 2DFA and $x$ is an input for $M$ of length $n$, then a tape head position for $M$ on $x$ is a number between 0 and $n + 1$ that indicates which cell the tape head is on. (0 indicates that the tape head is on the blank to the left of the input and $n + 1$ indicates that the tape head is on the blank to the right of the input.) A configuration for $M$ on $w$ is a quadruple $(w, p, i, j)$ where $p$ is the current state of $M$, $i$ is the position of the first tape head and $j$ is the position of the second tape head. Since $M$ cannot write on its tape, if $M$ ever repeats a configuration, then $M$ is in an infinite loop.

A Turing machine $N$ to decide $A_{2DFA}$ is given by

$N = \text{"On input } \langle M, x \rangle \text{ where } M \text{ is a 2DFA and } x \text{ is an input string:} $

1. Simulate $M$ on $x$ until either a) $M$ accepts, b) $M$ tries to move left from the initial blank, c) $M$ tries to move right from the final blank, or d) $M$ repeats a configuration.
2. If a) happens, then accept. If b), c) or d) happens, then reject.”

If $M$ is in an infinite loop, then since there are only finitely many configurations, $M$ must eventually repeat a configuration, so one of a), b), c), d) must eventually happen. Thus, $N$ decides $A_{2DFA}$.

(b) To show that $E_{2DFA}$ is undecidable, we will reduce $A_{TM}$ to $E_{2DFA}$. Given a Turing machine $M$ and an input $w$, we can construct a 2DFA $B$ such that $L(B)$ is the set of accepting configurations of $M$ on $w$. This $B$ is similar to the LBA $B$ constructed in the proof of Theorem 5.9, except that to check if $C_{i+1}$ follows from $C_i$ legally, this $B$ does not zig-zag back and forth using dots as markers, but instead it puts one tape head on $C_i$ and the other tape head on $C_{i+1}$ and compares the two without having to write anything on the tape.

Suppose that $R$ is a TM that decides $E_{2DFA}$. Then, the following TM $S$ decides $A_{TM}$.

$S = \text{"On input } \langle M, w \rangle \text{, where } M \text{ is a TM and } w \text{ is an input string:} $

1. Construct the 2DFA $B$ as described above.
2. Run $R$ on $\langle B \rangle$.
3. If $R$ rejects, then accept; if $R$ accepts, then reject.”

Since $A_{TM}$ is not decidable, there cannot be such a TM $S$, so there is not such TM $R$ and $E_{2DFA}$ is undecidable.

2. Exercise 5.4

No. For example, let $A = \{0^n1^n | n \geq 0 \}$ and $B = \{0^n1^m | n,m \geq 0 \}$. We will show that $A \leq_m B$ even though $A$ is not regular and $B$ is regular. A
mapping reduction \( f \) from \( A \) to \( B \) is computed by the Turing machine \( F \) given by

\[ F = \{ \text{On input } w \in \{0,1\}^*, \]

1. Determine if \( w \) belongs to \( A \). (Since \( A \) is decidable, \( F \) can do this with no further information.)

2. If \( w \) belongs to \( A \), then output 01. If \( w \) does not belong to \( A \), then output 10.

(In fact, the same argument shows that if \( A \) is any decidable language and \( B \) is any language other than \( \emptyset \) and \( \Sigma^* \), then \( A \leq_m B \).

3. **Problem 5.22**

   \( A \) is Turing-recognizable if and only if \( A \leq_m A_{TM} \).

   **Proof:** First suppose that \( A \) is Turing-recognizable and let \( M \) be a Turing machine that recognizes \( A \). The function \( f \) defined by \( f(w) = \langle M, w \rangle \) is a reduction from \( A \) to \( A_{TM} \) because it is obviously computable and we have

   \[ w \in A \text{ iff } M \text{ accepts } w \text{ iff } \langle M, w \rangle \in A_{TM} \text{ iff } f(w) \in A_{TM}. \]

   Now suppose that \( A \leq_m A_{TM} \). We know that \( A_{TM} \) is Turing-recognizable, so by Theorem 5.28, \( A \) is Turing-recognizable.

4. **Problem 5.23**

   \( A \) is decidable if and only if \( A \leq_m 0^*1^* \).

   **Proof:** First suppose that \( A \) is decidable. Define \( f \) by \( f(x) = 01 \) if \( x \in A \) and \( f(x) = 10 \) if \( x \notin A \). Since \( A \) is decidable, \( f \) is computable and \( x \in A \) if and only if \( f(x) \in 0^*1^* \), so \( A \leq_m 0^*1^* \).

   Conversely, suppose that \( A \leq_m 0^*1^* \). Since \( 0^*1^* \) is decidable, \( A \) is decidable by Theorem 5.22.

5. **Problem 5.24**

   The set \( \overline{A_{TM}} \) is mapping reduced to \( J \) by the function \( f(y) = 1y \). Thus, \( J \) is not Turing-recognizable. The set \( A_{TM} \) is mapping reduced to \( J \) by the function \( g(x) = 0x \). This shows that \( A_{TM} \) is mapping reducible to \( J \) and hence that \( J \) is not Turing-recognizable.

6. **Problem 5.25**

   Consider the set \( J \) of Problem 5.24. According to that problem, \( J \) is not Turing-recognizable, so \( J \) is not decidable. We will show that \( J \leq_m J \), so \( B = J \) is a solution to the problem.

   First note that \( J = \{ w | w = 0x \text{ for some } x \in \overline{A_{TM}} \text{ or } w = 1y \text{ for some } y \in A_{TM} \text{ or } w = \varepsilon \text{ or } w \text{ begins with a symbol other than 0 or 1} \} \). Let z₀
be some fixed string in \( J \), for example, \( z_0 \) could be \( 0x_0 \) for some particular \( x_0 \) in \( A_{TM} \). Define \( f : \Sigma^* \rightarrow \Sigma^* \) by

\[
f(w) = \begin{cases} 
1x & \text{if } w = 0x \\
0y & \text{if } w = 1y \\
z_0 & \text{if } w = \varepsilon \text{ or } w \text{ starts with a symbol other than } 0 \text{ or } 1.
\end{cases}
\]

Then, it is clear that \( f \) is computable. To see that \( f \) mapping reduces \( J \) to \( \overline{J} \), suppose first that \( w \in J \). We must show that \( f(w) \in \overline{J} \). If \( w \in J \), there are two possibilities. If \( w = 0x \) with \( x \in A_{TM} \), then \( f(w) = 1x \) with \( x \in A_{TM} \), so \( f(w) \in \overline{J} \). If \( w = 1y \) with \( y \in \overline{A_{TM}} \), then \( f(w) = 0y \) with \( y \in \overline{A_{TM}} \), so \( f(w) \in \overline{J} \). Thus, if \( w \in J \), then \( f(w) \in \overline{J} \).

Now suppose that \( w \notin J \). We must show that \( f(w) \notin \overline{J} \). There are four possibilities to consider. If \( w = 0x \) with \( x \in \overline{A_{TM}} \), then \( f(w) = 1x \), so \( f(w) \notin \overline{J} \). If \( w = 1y \) with \( y \in A_{TM} \), then \( f(w) = 0y \), so \( f(w) \notin \overline{J} \). If \( w = \varepsilon \) or \( w \) starts with a symbol other than 0 or 1, then \( f(w) = z_0 \), so \( f(w) \notin \overline{J} \). Thus, if \( w \notin J \), then \( f(w) \notin \overline{J} \).

This shows that \( f \) is a mapping reduction of \( J \) to \( \overline{J} \).

7. It is not possible to \( m \)-reduce \( \text{ALL}_{CFG} \) to \( \text{ALL}_{DFA} \).

**Proof:** Suppose that \( \text{ALL}_{CFG} \leq_m \text{ALL}_{DFA} \). Then by Exercise 4.3, \( \text{ALL}_{DFA} \) is decidable, so by Theorem 5.22, \( \text{ALL}_{CFG} \) is decidable. This contradicts Theorem 5.13.

8. It is not possible to \( m \)-reduce \( \text{EQ}_{TM} \) to \( \overline{A_{TM}} \).

**Proof:** Suppose that \( \text{EQ}_{TM} \leq_m \overline{A_{TM}} \). Then we have \( \text{EQ}_{TM} \leq_m \overline{A_{TM}} \). By Theorem 5.30, \( \text{EQ}_{TM} \) is not Turing recognizable, so by Corollary 5.29, \( A_{TM} \) is not Turing recognizable. This contradicts the fact that \( A_{TM} \) is Turing recognizable, as stated on page 202 of the text.