Problem 1

(a) $J = \{ w \mid w = 0x$ for some $x \in A_{TM}$ or
$w = 1y$ for some $y \in \overline{A}_{TM} \}$

Here's a reduction from $\overline{A}_{TM}$ to J.

Given $\langle M, w \rangle \in \overline{A}_{TM}$ let $f(\langle M, w \rangle) = 1\langle M, w \rangle$.

Clearly $\langle M, w \rangle \in \overline{A}_{TM} \iff f(\langle M, w \rangle) \in J$.

We can also reduce $\overline{A}_{TM}$ to $\overline{J}$. Here, given $\langle M, w \rangle \in \overline{A}_{TM}$, let $g(\langle M, w \rangle) = 0\langle M, w \rangle$. Again

$$\langle M, w \rangle \in \overline{A}_{TM} \iff g(\langle M, w \rangle) \in \overline{J}.$$

Therefore, neither J nor $\overline{J}$ is Turing recognizable.

(b) Give an example of an undecidable language B such that $B \leq_m \overline{B}$.

Obviously neither B nor $\overline{B}$ can be Turing recognizable because otherwise both will be and, therefore, B will be decidable. The example of J above works.

Given $w = 0x \in J$, let $f(w) = 1x \notin J$
also $w = 1y \in J$, let $f(y) = 0y \notin J$.

So $J \leq_m J$.

Problem 2

Testing for useless state:

In NFA.

For every $q \in Q$, make q an accept state, and all other states non-accept states. Then check if the NFA has an empty language. If it does, This means q is never reached on any input.

IN PDA

Same as above, and in addition, when q is entered, empty the stack. Then check if PDA has an empty language.

IN Turing machines

This is undecidable. Given Turing machine M and input M, construct the following machine D.

$D =$ "on any input x:

1. Run M on w
2. If M accepts, enter a special state $q_{special}$; otherwise if M rejects, halt."

Transitions from $q_{special}$ can be anything. Now

$$M \text{ accepts } w \iff D \text{ enters } q_{special}.$$

So $q_{special}$ is useless if M does not accept w. Being able to test for that can decide A_{TM}.

Problem 3

To test whether a Turing machine M on input w ever attempts to move it's head left when it's on the left-most tape cell is undecidable.

D = "on input $\langle M, w\rangle$:

1. Mark the left-most tape cell using a special symbol \$

2. Simulate M on w using the rest of the tape, and whenever \$ is reached (it means M moved beyond it's tape) move right.

3. If M accepts w, move left until \$ is reached, and then attempt to move left."

D attempts to move beyond the left-most cell if and only if M accepts w. So we can decide A_{TM}.

The other problem is decidable because if you Run M on w for a long enough time and it makes no left moves, then it will never move left. long enough can be upper bounded by $|w| + |Q|$ which is enough to traverse the input and then keep moving right.

Problem 4

Show that PCP is decidable when $\Sigma = \{1\}$.

What's relevant is the length of the strings on top and bottom of a domino. Given a domino, let the value of the domino be defined as the length of top string minus the length of bottom string. Then there is a match iff

1. There exists a domino with value 0

or 2. there exist two dominos, one with positive value and one with negative value.

This can be decided easily. Now we prove it.

$\Rightarrow$ If there is a match, and no domino has value 0, then they can't be all positive or all negative.

$\Leftarrow$ If ~~[illegible]~~ (1) is satisfied, that domino gives the match.

If (2) is satisfied, then let the positive and negative values be a and b. Then repeat the first domino $-b$ times and the second a times. This gives a match.

Problem 5

Let M_1 be a Turing machine such that $L(M_1) = \phi$. If $M_1 \in P$, $\exists\ M_2 \notin P$. If $M_1 \notin P$, $\exists\ M_2 \in P$. So M_1 and M_2 have different membership in P.

Now given M and w, construct:

$D =$ "on input x:

1. Run M on w
2. If M accepts w, Run M_2 on x
3. otherwise, reject x."

M accepts $w \Leftrightarrow L(D) = L(M_2)$

M does not accept $w \Leftrightarrow L(D) = L(M_1) = \phi$

where $L(M_2) \neq L(M_1)$

Testing membership for D in P can decide whether M accepts w. So P is undecidable.