Name: Kevin Zhang

Problem 1.

Assume that E_{TM} is turing recognizable. From the previous homework, problem 10 shows that $\overline{E_{TM}}$ is Turing recognizable. From a theorem shown in class, we know that if $\overline{E_{TM}}$ is recognizable AND E_{TM} is recognizable, then E_{TM} is decidable. From the previous homework, problem 6c shows that E_{TM} is not decidable. Therefore, E_{TM} cannot be recognizable, as that would create a contradiction.

Problem 2.

We can build a decider for S as such:

For a given input $w \in \Sigma^*$, if w is not an encoding $\langle M \rangle$, then reject.

If it is an encoding $\langle M \rangle$, then do the following:

Construct DFA $\hat{M} \mid L(\hat{M}) = L(M) \cap L(M)^R$. This is possible because $L(M)$ is a regular language (because M is a DFA), and regular languages are closed under intersection and reverse.

Decide whether $L(\hat{M}) = L(M)$ (Equality problem for DFAs). If yes, accept. If no, reject.

Problem 3.

We can build a decider for E as such:

For a given input $w \in \Sigma^*$, if w is not an encoding $\langle M \rangle$, then reject.

If it is an encoding $\langle M \rangle$, then do the following:

Construct PDA $\hat{M} \mid L(\hat{M}) = L(M) \cap L(A)$, where $A = \{w \in \Sigma^* \mid w \text{ has more 1s than 0s}\}$. This is possible because $L(A)$ is a CFL (from homework 9, problem 1), and $L(M)$ is a regular language (because M is a DFA). The intersection of a CFL and a Regular Language is a CFL (from a theorem shown in class).

Decide whether $L(\hat{M}) = \varnothing$ (Emptiness problem for CFGs). If yes, reject. If no, accept.

Problem 4.

(a) We can construct $\hat{M}$ from M by doing the following:

Every transition $(q, a), (r, \sqcup, M)$, replace $\sqcup$ with a new tape symbol $x \mid x \notin \Gamma$. Then, add transitions $(q, x)(r, y, M)$ for every transition $(q, \sqcup)(r, y, M)$.

This effectively uses a new symbol x in lieu of the blank, but also treats x as if it were a blank.

(b) Let $K = \{\langle M \rangle \mid \text{M is a TM that on input} \varepsilon\text{, prints a blank}\}$.

$ACCEPT - EMPTY \leq_T K$ can be shown below:

We can construct a decider for $ACCEPT - EMPTY$ using an oracle for K. On input $w \in \Sigma^*$, if w is not encoding, reject. If w is encoding of M, then construct $\hat{M}$ which will print a blank on any input (ie. write a blank before the first letter) and then do what M does. From (a), we know that $L(\hat{M}) = L(M)$. Ask oracle $\langle \hat{M} \rangle \in K$. If yes, accept. If no, reject.

Therefore, K is undecidable, as $ACCEPT - EMPTY$ is undecidable.

Problem 5.

Let $J = \{\langle M, S\rangle \mid \text{M is a TM with some unreachable states } S\}$.

$E_{TM} \leq_T J$ can be shown below:

We can construct a decider for E_{TM} using an oracle for J. On input $w \in \Sigma^*$, if w is not encoding, reject. If w is encoding of M, then we can pass $\langle M, \{q_{accept}\}\rangle$ to oracle J. If oracle accepts, then we can reject, If oracle rejects, then we can accept. This because if q_{accept} is unreachable, then $L(M) = \emptyset$. If it is reachable, then there are some words accepted, so $L(M) \neq \emptyset$.

Therefore, J is undecidable, as E_{TM} is undecidable.

Problem 6.

Formally, we can define $f : \Sigma^* \to \Sigma^*$ as follows:

$$f(x) = \begin{cases} x & \text{if} x \notin A \\ 1111\hat{x} & \text{where} \hat{x} = x \text{ with beginning 0 removed} \end{cases}$$

f in this case creates a mapping for all elements in A to a similar element in B by replacing the 0 in beginning with 1111. Thus, $A \leq_M B$.

Problem 7.

For each language $L \subseteq \Sigma^*$, we want to show L is enumerable $\Rightarrow L \leq_m A_{TM}$.

For an arbitrary language L, if L is enumerable, that means there is an enumerator E that prints $w_0, w_1, w_2, \ldots \in L$. Using this enumerator, we can make a function f such that $f(w) = \langle L, w\rangle$, for all w enumerated by E.

Formally, we can define $f : \Sigma^* \to \Sigma^*$ as follows:

$$f(x) = \begin{cases} x & \text{if x is not enumerated by } E \\ \langle L, x\rangle & \text{if x is enumerated by} E \end{cases}$$

This f applies to all enumerable languages L.