Problem 1 Use a reduction to show that the language ALL_{TM} is undecidable

$$ALL_{TM} = \{\langle M \rangle \mid \text{where } M \text{ is a TM and } L(M) = \Sigma*\}$$

Lets suppose that a turing machine, R, decides ALL_{TM}. Using this assumption, lets show how to decide the acceptance problem, A_{TM}. On the input $\langle M, w \rangle$, construct M_1 as follows:

$M_1 =$ "On input string x, simulate M on w:

1. If M accepts, *accept.*
2. If M rejects, *reject."*

Note, that if M accepts w, $L(M_1) = \Sigma*$. Otherwise, $L(M_1) = \emptyset$. This ensures that the TM we are about to construct will only deal with the empty language or the "all" language. Now, we can define TM S as follows:

$S =$ "On input $\langle M, w \rangle$, construct M_1 as shown above.

1. Run R on input $\langle M_1 \rangle$. If R accepts, *accept.*
2. Otherwise, *reject."*

Here we have shown that if we assume there exists some R that can decide ALL_{TM}, we can construct a reduction that would make A_{TM} decidable. Since A_{TM} was proven undecidable (using diagonalization), there cannot exist an R that decides ALL_{TM}. Therefore, it is undecidable.

Problem 2 A *useless state* in a Turing machine is one that is never entered on any input string. Consider the problem of determining whether a Turing Machine has any useless states. Formulate this problem as a language and show that it is undecidable.

Solution We should formulate the language as follows:

$$USELESS_{TM} = \{\langle M, q \rangle \mid \text{where } M \text{ is a TM and } q \text{ is a state in } M\}$$

Suppose the turing machine R decides USELESS. R takes a Turing machine and a state as input and determines if that state is useless. So, give R the input $\langle M, q_{accept} \rangle$ so it can determine if the accept state of M is useless. If it is, the language is empty. Lets define S as the turing machine that will decide E_{TM}, which was proven in the book to be undecidable. Give S the input M, and run R on $\langle M, q_{accept} \rangle$.
If R accepts, *accept.*
If R rejects, *reject.*

Therefore, using the assumption that USELESS is decidable, we found a reduction from E_{TM} to USELESS showing that E_{TM} is decidable. However, E_{TM} was already proven to be undecidable, so our assumption is false and we can conclude that R does not exist.

Problem 3 If $A \leq_m B$ and B is a regular language, does this imply that A is a regular language? Why or why not?

Solution No it does not. The language $0^i 1^i$ is not regular, but it can be mapped to the language, $0^i 1^k$, which is regular.

The reduction function is as follows: $f(w) = 0^i 1^i$ if $w \in A$ or $f(w) = 10$ if $w \notin A$.

This way, $w \in A$ iff $f(w) = 0^i 1^i$ and $f(w) \in B$, where $i \geq 0$.

This shows that B is a regular language, but A is not.
The language A is a Context Free Language, so it is also turing decidable (theorem 4.9). Therefore, we can conclude that f is a computable function.

Problem 4 Prove that the language

$$LOOP_{TM} = \{\langle M \rangle \mid M \text{ is a TM and } M \text{ loops on all inputs}\}$$

is not recognizable.

Solution First, assume $LOOP_{TM}$ is recognizable. We will show this assumption to be false later.
$\overline{LOOP}_{TM}$ can be described as follows:

$$\overline{LOOP}_{TM} = \{\langle M \rangle \mid M \text{ is a TM and } M \text{ halts on at least one input}\}$$

We can build a recognizer for this turing machine by non-deterministically running every input until the TM halts for at least one of them. By definition, this makes it a recognizer.

So far, we have assumed $LOOP_{TM}$ is turing recognizable and we have shown $\overline{LOOP}_{TM}$ is also turing recognizable. From this, we can conclude that both are decidable (theorem 4.22).

In order to prove our original assumption false, we must show a reduction from a currently known undecidable TM to either $LOOP_{TM}$ or $\overline{LOOP}_{TM}$ to show that one of them is undecidable. We will show that $LOOP_{TM}$ is undecidable.

Let R be a decider for $LOOP_{TM}$. Now let S be a decider for A_{TM}. Define it as follows:

$S =$ "On input $\langle M, w \rangle$:

1. Run R on M.
2. If R accepts, accept.
3. If R rejects, reject."

Here we have shown that A_{TM} is decidable using our assumption that $LOOP_{TM}$ is recognizable and therefore decidable. However, A_{TM} was previously proven to be undecidable by diagonalization in the textbook. So, $LOOP_{TM}$ is undecidable, and therefore unrecognizable.