1. Let $C$ be a language. Prove that $C$ is Turing-recognizable iff a decidable language $D$ exists such that $C = \{x | \exists y ((x, y) \in D)\}$

Solution:
We need to prove both directions. To handle the easier one first, assume that the decidable language $D$ exists. A TM recognizing $C$ operates on input $x$ by going through each possible string $y$ and testing whether $(x, y) \in D$. If such a $y$ is ever found, accept; if not, just continue searching.

For the other direction, assume that $C$ is recognized by TM, denoted by $M$. Define the language $D$ to be $\{(x, y) \mid M \text{ accepts } x \text{ within } |y| \text{ steps}\}$. Language $D$ is decidable since one can run $M$ for $y$ steps and accept iff $M$ has accepted. If $x \in C$, then $M$ accepts $x$ within some number of steps, so $(x, y) \in D$ for some sufficiently long $y$, but if $x \notin C$ then $(x, y) \notin C$ for any $y$.

2. Let $T = \{(M) \mid M \text{ is a TM that accepts } w^R \text{ whenever it accepts } w\}$ ($w^R$ is the reverse of $w$). Show that $T$ is undecidable. Is $T$ recognizable?

Solution:
$T$ is unrecognizable. We prove it by contradiction. Assume $M_T$ is a recognizer of $T$, we can construct a recognizer of $\overline{A_{TM}}$.

$M' = $ “On input $(M, w)$:

1. Construct a TM $M_w$: “On input x:

   (a) If x=$w^r$, REJECT;
   (b) If x=$w$, run $M$ on $w$ and return the same result;
   (c) Else, REJECT.”

2. Run $M_T$ on $(M_w)$, return the same.”

Correctness:
$M'$ accepts $(M, w) \iff M_T$ accepts $(M_w) \iff M$ does not accept $w \iff (M, w) \in \overline{A_{TM}}$.
Therefore, $T$ is unrecognizable.

3. Let

$$M = \{(a, b, c, p) : a, b, c \text{ and } p \text{ are binary integers, such that } a^b \equiv c \mod p\}$$

Show that $M \in P$.

Solution:
The most obvious algorithm multiplies $a$ by itself $b$ times then compares the result to $c$, modulo $p$. That uses $b-1$ multiplications and so is exponential in the length of $b$. Hence this algorithm doesn’t run in polynomial time. Here is one that does:

“On input $(a, b, c, p)$, four binary integers:
• Let \( r := 1 \).
• Let \( b_1, \ldots, b_k \) be the bits in the binary representation of \( b \).
• For \( i := 1, \ldots, k \):
  – If \( b_i = 0 \), let \( r := r^2 \mod p \).
  – If \( b_i = 1 \), let \( r := ar^2 \mod p \).
• If \( (c \mod p) = (r \mod p) \), accept; otherwise reject.”

The algorithm is called repeated squaring. Each of its multiplications and modular operations can be done in polynomial time in the standard way, because the \( r \) is never larger than \( p \). The total number of multiplications is proportional to the number of bits in \( b \). Hence the total running time is polynomial.

4. Prove that the following language is undecidable:

\[ A = \{ \langle M \rangle : M \text{ is a TM with running time } O(n) \} \]

Solution:
We give a mapping reduction from \( H_{TM} \) to it as follows.
Given an instance \( \langle M, w \rangle \) of \( H_{TM} \), construct the following TM \( M_0 \):

\[ M_0 = \text{“On input } x \text{ of length } n:\]

• Ignore \( x \)
• Run \( M \) on \( w \).
• If \( M \) accepts \( w \), accept.
• If \( M \) rejects \( w \), reject.

The reduction is correct: If \( M \) halts on \( w \), then the running time of \( M_0 \) is a constant, and \( O(n) \) in particular. If \( M \) loops on \( w \), then \( M_0 \) loops on every input and is not in \( A \).

ALTERNATIVE solution, mapping reduction from \( A_{TM} \).

\[ M_1 = \text{“On input } x \text{ of length } n:\]

• Ignore \( x \)
• Run \( M \) on \( w \).
• If \( M \) accepts \( w \), accept.
• If \( M \) rejects \( w \), loop.

\( \langle M, w \rangle \in A_{TM} \iff M_1 \text{ accepts } \iff M_1 \text{ runs in constant time } \Rightarrow M_1 \in A. \)

5. Prove that the following language is undecidable:

\[ A = \{ \langle M \rangle : L(M) \in \text{TIME}(n) \} \]

Solution:
\[ L_1 = \{1\}^* \in \text{TIME}(n) \implies A \neq \emptyset \]
\( A_{TM} \) is not decidable. Let \( R \) be its recognizer, clearly its running time is infinite \( \implies \langle R \rangle \notin A \implies A \neq \{ \langle M \rangle \} \). Hence the property is non-trivial. It is clearly a property of the language of \( M \). By Rice’s theorem, we know that \( A \) is undecidable.