Prove that the language $L$ defined below is undecidable. Is it recognizable, why or why not?

$$L = \{ \langle M, w \rangle \mid M \text{ rejects the input } w \}$$

Solution: We show that if $L$ is decidable, $A_{TM}$ is also decidable.

Suppose, for the sake of contradiction, that $L$ is decidable. Then some machine $D_L$ decides it. We use $D_L$ to construct a decider for $A_{TM}$.

$M_{ATM}$ on input $\langle M, w \rangle$:
1. Construct $M_{\langle M, w \rangle}$
2. Run $D_L$ on $\langle M_{\langle M, w \rangle}, 1 \rangle$
   - If $D_L$ accepts, ACCEPT
   - If $D_L$ rejects, REJECT

where $M_{\langle M, w \rangle}$ is defined as follows

$M_{\langle M, w \rangle}$ on input $x$:
1. Run $M$ on $w$
   - if $M$ accepts, REJECT
   - if $M$ rejects, LOOP

Since $M_{ATM}$ and $M_{\langle M, w \rangle}$ are finite, and each of the instructions is computable, both are valid TM specifications.

Suppose $\langle M, w \rangle \in A_{TM}$. Then $M$ accepts $w$. So $M_{\langle M, w \rangle}$ rejects on all inputs, specifically on input ‘1’. So $D_L$ accepts $\langle M_{\langle M, w \rangle}, 1 \rangle$ and $M_{ATM}$ accepts.

Suppose $\langle M, w \rangle \notin A_{TM}$. Then $M$ does not accept $w$, and so either rejects or loops on $w$. In either case, $M_{\langle M, w \rangle}$ loops on all inputs, specifically on input ‘1’. So $D_L$ rejects $\langle M_{\langle M, w \rangle}, 1 \rangle$ and $M_{ATM}$ rejects.

Therefore, $M_{ATM}$ decides $A_{TM}$. Since $A_{TM}$ is undecidable, our supposition is false and the result is proved.

We construct the following machine to recognize $L$.

$R_L$ on $\langle M, w \rangle$:
1. Run $M$ on $w$
   - If $M$ rejects, ACCEPT
   - If $M$ accepts, REJECT

Since each of $R_L$’s instructions is a computable procedure, $R_L$ is a TM.

If $\langle M, w \rangle \in L$, then $M$ rejects $w$, so $R_L$ accepts. If $\langle M, w \rangle \notin L$, then $M$ does not reject $w$ and so either accepts or loops. In either case, $R_L$ does not accept. Therefore, $R_L$ recognizes $L$ and $L$ is recognizable.