1. A standard Turing Machine (TM) has an input alphabet $\Sigma$ and has input strings $x$ which are finite strings of symbols from $\Sigma$.

   $M(x)$ = the result of TM $M$ on input $x$ = either "accept" or "reject" or "loop". (Here loop means "never halt" and the TM $M$ may either actually loop on a finite part of its tape or run forever moving farther and farther along on its tape.)

2. The TM $M$ recognizes a language $L(M)$ defined by, $L = \{ x | x \text{ is an input string of } M \text{ and } M(x) \text{ accepts} \}$. Given a TM $M$, the language $L(M)$ is a subset of $\Sigma^*$ and is unique.

3. The language $L$ is decidable if there is some TM $M$ which recognizes $L$ and which halts on every legal input string $x$ for $M$. In this case we say $M$ accepts $x$ if $x \in L$ (i.e. "$x$ is in $L$"), and otherwise it rejects $x$.

4. A language $L$ is enumerable if there is a Turing machine with a printer which enumerates (lists) the elements of $L$. $L$ is enumerable if and only if it is recognizable (Theorem 3.21).

5. A language is undecidable if there is no TM which decides it. A language is unrecognizable if there is no TM which recognizes it.

6. Recall that a TM that decides a language $L$ also recognizes $L$. It follows from this that if $L$ is unrecognizable then $L$ is undecidable.

7. The language $A_{TM} = \{ < M, w > | M \text{ is a TM that accepts the input string } w \}$. $A_{TM}$ is a language which is recognizable but $A_{TM}$ is not decidable. In fact, $A_{TM}$ is recognized by a UTM, a universal Turing Machine. UTM is not unique.

From this it follows that the complement of $A_{TM} = \{ < M, w > | M \text{ is a TM and } w \text{ is a legal input string for } M \text{ and } M \text{ does not accept the input string } w \}$ in not even recognizable.