Give an implementation level TM for the following
\[ \mathcal{L} = \{w : w \text{ contains a number followed by that many } a\text{'s}\} \]
Ex: 1010aaaaaaaaaa

\[
S = "On input w:
1. Scan through w and check a's to the right of numbers, if numbers found after a or a's found before numbers reject.
2. Move left until number reached.
3. If 0 mark and go to step 4, if 1 mark current position with # go right and unmark all numbers. Go to rightmost number and mark it then go right and mark an "a". Go left until unmarked number found. Mark current number then unmark all numbers right and repeat process of going right and marking a's then returning until all marked including #. If run out of a's to mark, reject.
4. Return to start, scan through w and check if all a's marked, if all marked and numbers remain reject, if everything marked accept.
5. Move to end of string then return to step 2."
\]

Reducibility(S215)(M169)
- Method for proving a problem is computationally as hard as another problem
- A reduction is converting one problem to another, and convecting the solution back.
- Involves two problems \( \Rightarrow \) If A reduced to B we can use the solution to B to solve A.

\[ A \leq_r B \]
Always reduce known to unknown

\[ \text{HALT}^\text{TM} = \{<M,W>: M \text{ is a TM that halts on input } w\} \]

\[ A^\text{TM} \leq \text{HALT}^\text{TM} \text{ - Reduce } A^\text{TM} \text{ to } \text{HALT}^\text{TM} \]

Thm. \(\text{HALT}^\text{TM}\) is undecidable
proof assume \(\text{HALT}^\text{TM}\) is decidable

1. Let \(R\) be a decider for \(\text{HALT}^\text{TM}\)
2. Construct TM \(S\) to decide for \(A^\text{TM}\)

\[ S = \text{"On input } <M,W> \text{ M is an encoding of a TM and W is a string:}\]
\[ \text{CA}^\text{TM} = \{<M,W>: \text{This } A^\text{TM} \text{ that accepts } W\} \]
1. Run \(R\) on input \(<M,W>\).
2. If \(R\) rejects, reject.  
3. If \(R\) accepts, simulate \(M\) on \(W\) until it halts because we know it will.
4. If \(M\) accepts, accept, if \(M\) rejected, reject."

Thus, if \(R\) decides \(\text{HALT}^\text{TM}\), then \(S\) decides \(A^\text{TM}\) since \(A^\text{TM}\) is undecidable, our assumption was wrong and \(\text{HALT}^\text{TM}\) is undecidable.

\[ \text{ET}^\text{TM} = \{<M>: M \text{ is a } \text{TM and } L(M) = \emptyset\} \]

\[ A^\text{TM} \leq \text{ET}^\text{TM} \text{ - Reduce } A^\text{TM} \text{ to } \text{ET}^\text{TM} \text{ Thm. } \text{ET}^\text{TM} \text{ is undecidable proof, assume } \text{ET}^\text{TM} \text{ is decidable}\]

let \(R\) be a decider for \(\text{ET}^\text{TM}\) construct TM \(S\) to decide \(A^\text{TM}\)

Construct a TM \(S\) that modifies \(M\) to reject all strings except \(W\)

\[ M_1 = \text{"On input } x: \]
1. If \(x\neq w\), reject
2. If \(x=w\) run \(M\) on input \(w\) and accept if \(M\) does

\[ S = \text{"On input } <M,W> \text{ M is an encoding of a TM and W is a string:}\]
1. Use the description of \(M\) and \(w\) to construct \(M_1\) as defined.
2. Run \(R\) on input \(<M>\)
3. If \(R\) accepts, reject, if \(R\) rejects, accept
If R is a decider for $E_{TM}$, S would be a decider $A_{TM}$ since $A_{TM}$ is undecidable, $E_{TM}$ must be undecidable.

$E_{TM} = \{<M_1, M_2>: M_1$ and $M_2$ are $TM$'s and $L(M_1) = L(M_2)\}$

Thm. $E_{A_{TM}}$ is undecidable

Proof. Assume $E_{A_{TM}}$ is decidable let $TM$ R decide $E_{Q_{TM}}$, and construct a $TM$ S to decide $E_{TM}$

S = “On input $<M>$, where M is a $TM$

1. Construct $TM$ M, that rejects all inputs
2. Run R on $<M, M_2>$
3. If R accepts, accept if R rejects, reject”