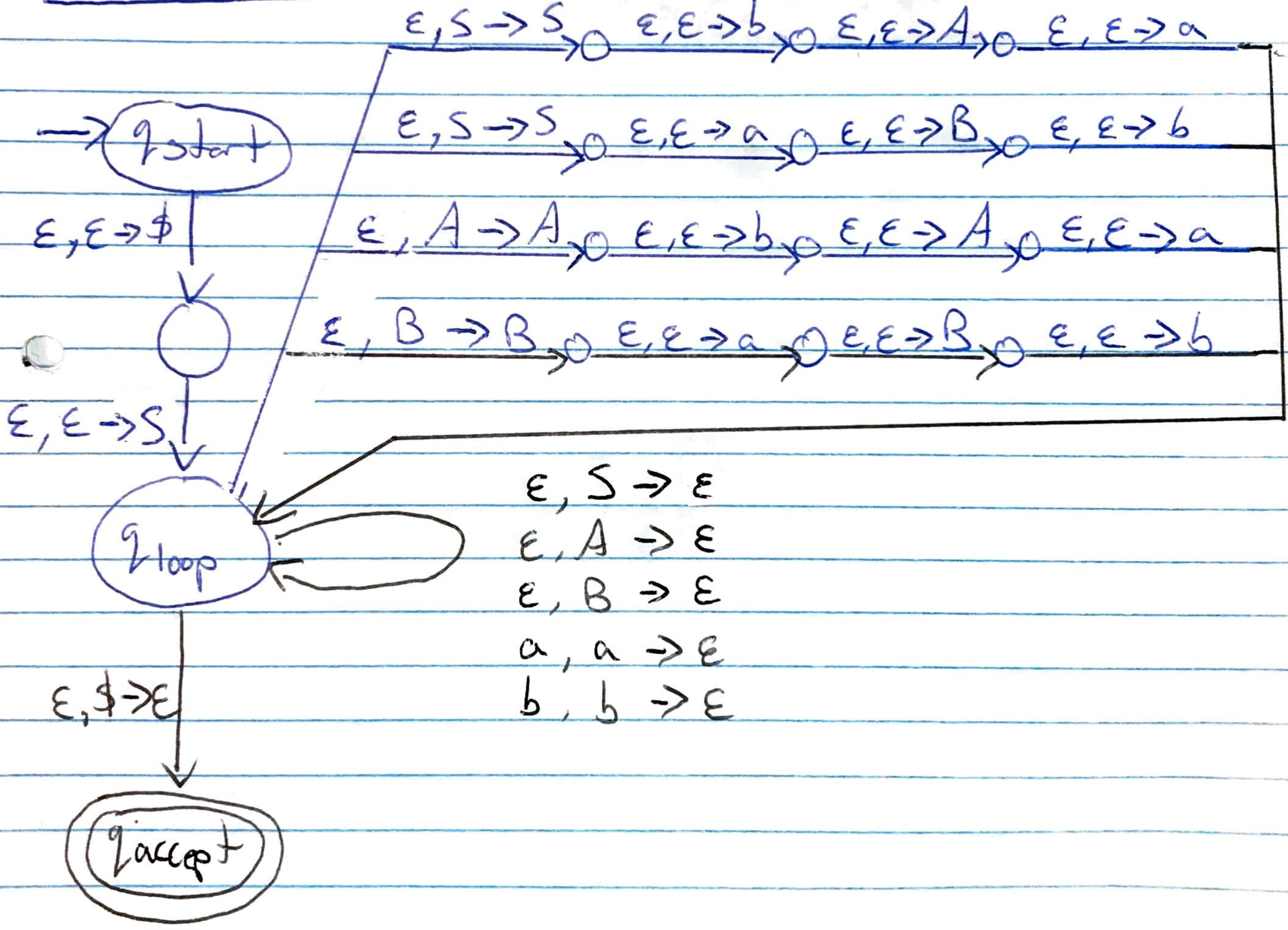
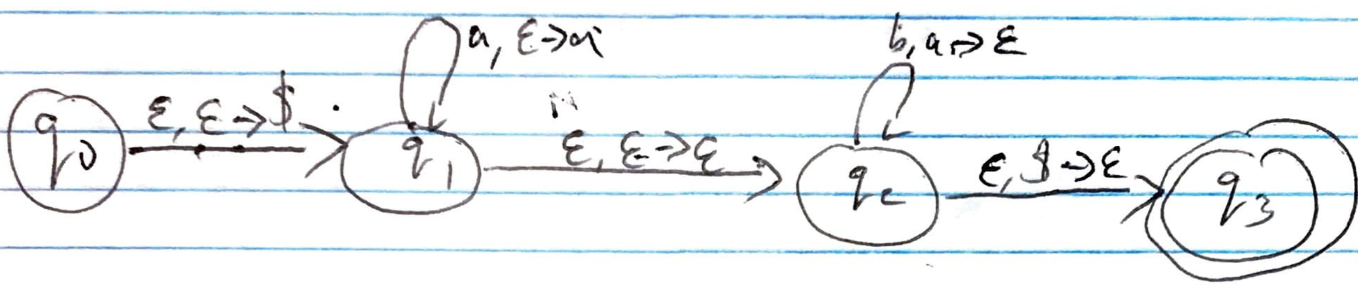
1. Convert the following CFG to a PDA using the procedure described in class:

$S \to aAbS \mid bBaS \mid \varepsilon$
$A \to aAbA \mid \varepsilon$
$B \to bBaB \mid \varepsilon$



2. The following PDA recognizes the language $L = \{a^n b^n \mid n \geq 0\}$. Convert the PDA to CFG.

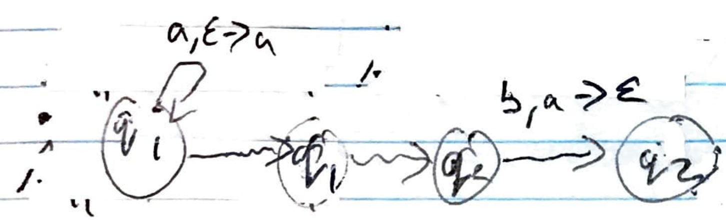


- **Variables**: $A_{00}, A_{11}, A_{22}, A_{33}, A_{01}, A_{02}, A_{03}, A_{12}, A_{13}, A_{23}$

- **Start Variable**: A_{03}

- **Productions**: $A_{03} \to A_{01} A_{13}$
  $A_{13} \to A_{12} A_{23}$

$A_{00} \to \varepsilon$
$A_{11} \to \varepsilon$
$A_{22} \to \varepsilon$
$A_{33} \to \varepsilon$
$A_{01} \to \varepsilon$
$A_{23} \to \varepsilon$



- $A_{12} = a A_{12} b \mid \varepsilon$

Therefore the CFG is

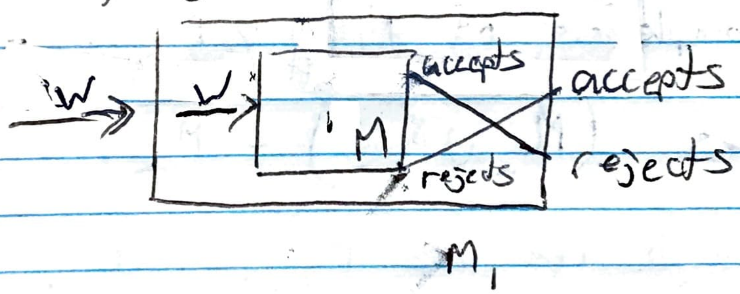
$S \to A_{03}$
$A_{03} \to A_{12} \mid \varepsilon$
$A_{12} \to a A_{12} b \mid \varepsilon$

2. Show that the class of decidable languages is closed under
a) complementation, b) concatenation, c) intersection,

1. For any two decidable languages L_1 and L_2, let M_1 and M_2 be the TM's that decide them.

a) We construct a TM M' that decides the complement of M

M' = "On input w:
1) Run M on w. If it accepts, reject. Otherwise, reject

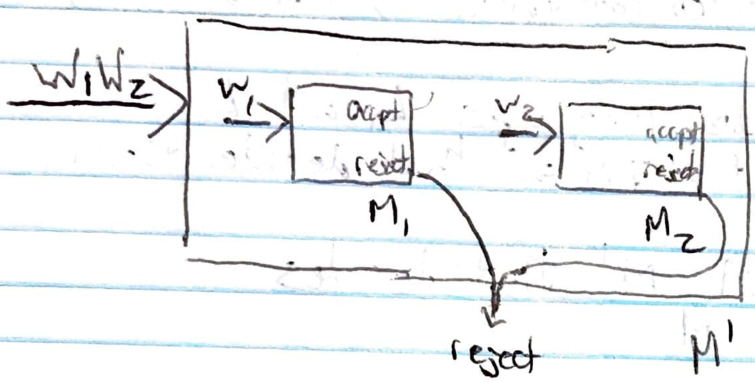


- Since M' does the opposite of M, we see the class of decidable languages are closed under complementation.

b) We construct a TM M' that decides the concatenation of L_1 & L_2, where TM's M_1 and M_2 decide L_1 and L_2 respectively.

M' = "On input w where $w = w_1 w_2$
1) Run M_1 on w_1. If it rejects, M' rejects
2) Run M_2 on w_2. If it rejects, M' rejects.

3) If both M_1 and M_2 accepted, then accept.



- If either M_1 or M_2 reject M' rejects. This can be written as: $M_{1_{reject}} \cup M_{2_{reject}}$

↳ The complement of $M_{1_{reject}} \cup M_{2_{reject}}$ by deMorgan's law is $\overline{(M_{1R} \cup M_{2R})} = \overline{M_{1R}} \cap \overline{M_{2R}} = \overline{Reject}$

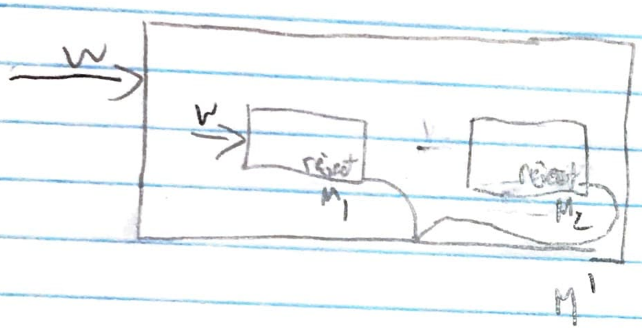
↳ This can be read as $M_{1_{accept}} \cap M_{2_{accept}} = Accept$

since a decider will either accept or reject.

↳ Therefore the concatenation of L_1 & L_2 (decidable languages) is proven to be decidable.

c) We construct a TM M' that will decide the intersection of L_1 & L_2

M = "On input w:
1) Run M_1 on w. If M_1 rejects then reject.
2) Run M_2 on w. If M_2 rejects then reject.
3) If neither M_1 nor M_2 rejected, then accept.



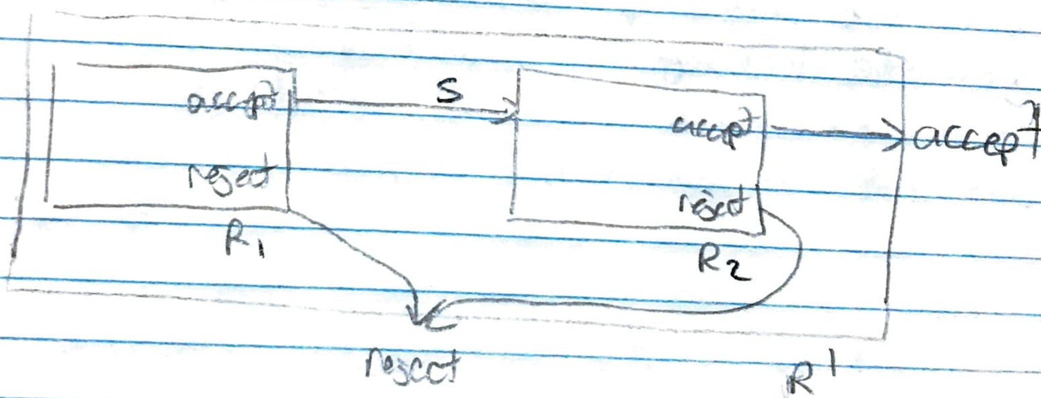
∴ Therefore we have proven that M' is able to decide $L_1 \cap L_2$, and we have proven decidable languages are closed under intersection.

7) Show that the class of recognizable languages is closed under: a) intersection and b) star.

a) Let R_1 be a recognizer of language L_1 and R_2 be a recognizer for L_2.
• We construct R' as a recognizer that recognizes the intersection of L_1 and L_2 where w is a string of the intersected language.

R' = " On input w

1) Run recognizer R_1 on w, if it accepts then we run R_2 on w. If it accepts then we accept. Else we reject.

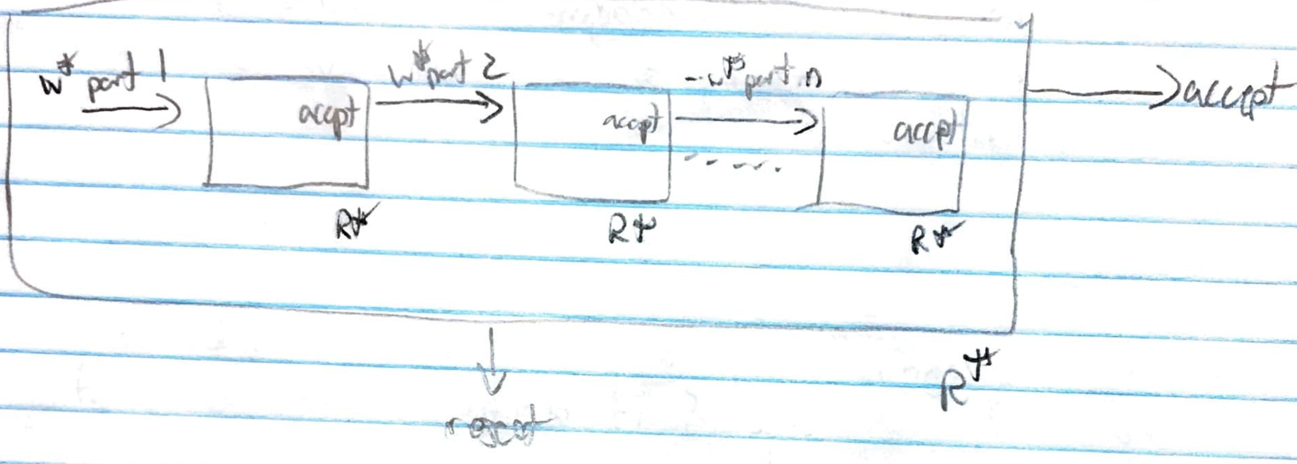


∴ Therefore we have proven that the class of recognizable languages is closed under intersection.

b) Let R_1 be a recognizer for the language L_1. We construct a recognizer R^* that will recognize L_1^*

$R^* =$ "On input w^* where each member of w^* is w with the kleene star operation attached to it.

1) Run each possible iteration of w^* through R^*, if they are all accepted, then accept, otherwise reject.



∴ Therefore we have proven that the class of recognizable languages is closed under star.

5. Consider the standard string order over the binary alphabet that we saw in class:

$\varepsilon, 0, 1, 00, 01, 10, 11, 000, 001, \ldots$

That is, strings are ordered in increasing order of length and lexicographical order. Show that a language is decidable iff some enumerator enumerates the language in standard string order.

- If an enumerator enumerates in standard string order then the decider can test each string as it appears, in which case the decider will either accept or reject. If it sees a string such as 001 and it was deciding for string 11, then it knows that 11 will never appear again, therefore it will reject.

- If a language is decidable then an enumerator can print out strings in lexigraphical order and test each one using the decider.

∴ As both directions have been proven we have shown a language is decidable iff some enumerator enumerates the language in standard string order.