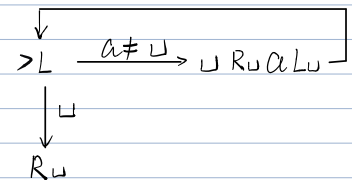
臧可 3180102095

Q1. $(q_0, \triangleright \underline{b}aba) \vdash_M (q_1, \triangleright \underline{a}aba) \vdash_M (q_0, \triangleright a\underline{a}ba) \vdash_M (q_1, \triangleright a\underline{b}ba)$
$\vdash_M (q_0, \triangleright ab\underline{b}a) \vdash_M (q_1, \triangleright aba\underline{a}) \vdash_M (q_0, \triangleright aba\underline{a}) \vdash_M (q_1, \triangleright aba\underline{b})$
$\vdash_M (q_0, \triangleright abab\underline{\sqcup}) \vdash_M (h, \triangleright abab\underline{\sqcup})$

Q2.



Q3. Proof:

$\Rightarrow$) Because A is recursive, $\bar{A}$ is recursive
Then A and $\bar{A}$ are recursively enumerable

$\Leftarrow$) Suppose that A is accepted by M_1 and $\bar{A}$ is accepted by M_2
Construct a new Turing Machine that decides A as follows:

1) run M_1, M_2 on input "w"
2) If M_1 accepts, accept
   if M_2 accepts, reject

$W \to M_1 \xrightarrow{\text{halts}} \text{accepts}$
$W \to M_2 \xrightarrow{\text{halts}} \text{rejects}$

Q4. Proof: Suppose that $\bar{H}$ is recursively enumerable.
Because H is recursively enumerable
by the statement in Q3 we can know that H is recursive. contradict
So $\bar{H}$ is not recursively enumerable

Q5. (a) False. Couterexample: H is recursively enumerable but $\bar{H}$ is not
(b) True.
(c) False. Couterexample: $>R \circlearrowleft \bar{a}$ this TM does not decide any language
(d) False. The couterexample is the same as (c)

Q6. Proof: Every language is a subset of Σ^*,
Σ^* is countable
As we have proved in the class that
any subset of a countable set is coutable.
every language is countable.

Q7. Define the language as $L = \{$"M""R" : M is a DFA and R is a regular expression, $L(M) = L(R)\}$

Suppose that a Turing machine M_1 decides the language
$EQ_{DFA} = \{$"A""B" : A and B are DFAs and $L(A) = L(B)\}$

Construst a Turing machie M_2 decides L:
on input "M""R" where M is a DFA and R is a regular expression
1. Convert R into a DFA D_R
2. Run M_1 on "M""D_R"
3. IF M_1 accepts, accept.
   IF M_1 rejects, reject.

so it is recursive.

Q8. Construct a TM M' that enumerates A as follows:

M' = a name all strings in Σ^* as $s_1, s_2, s_3, \ldots$ in lexicographically order

for $i = 1, 2, 3, \ldots,$ generate $s_1, s_2, \ldots, s_i$, for i steps

If any computation halts, accept

so A is recursively enumerable

Q9. Define the language as

$L = \{``M"``\sigma" :$ M writes the symbol σ at least once when started on the empty tape$\}$

Contruct a TM M_w and a symbol σ ($\sigma \notin \Sigma_M$, $\sigma \in \Sigma_{M_w}$)

on input x

1. erase x
2. write w ~~on its~~ tape
3. run M
4. write σ

then M halts on w iff M_w writes the symbol σ at least once when started on the empty tape

so L is not recursive. namely the problem is undecidable.