Theory of Computation, Fall 2023
Quiz 3 Solutions

Q1. In class, we have proved that EQ_{DFA} is recursive. Suppose Turing machine M_{EQ} decides

$$EQ_{DFA} = \{"M_1""M_2" : M_1 \text{ and } M_2 \text{ are two DFAs with } L(M_1) = L(M_2)\}.$$

To prove that S is recursive, it suffices to reduce S to EQ_{DFA}. We construct a Turing machine M_S that decides S as follows.

$M_S =$ on input $"M"$:
1. construct a DFA M_R with $L(M_R) = \{w^R : w \in L(M)\}$
2. run M_{EQ} on $"M""M_R"$
3. return the result of M_{EQ}

This completes the proof.

Q2. Let $L = \{"M" : "M" \text{ is a Turing machine that halts on some input}\}$. In class, we have proved that L is not recursive. To prove that A is not recursive, it suffices to reduce L to A. Suppose there is a Turing machine M_A decides A. Then we can construct a Turing machine M_L that decides L as follows.

$M_L =$ on input $"M"$:
1. construct a Turing machine M_{all} that halts on every input
2. run M_A on $"M""M_{all}"$
3. return the result of M_A

This completes the proof.

Q3. We show that A is recursively enumerable by presenting a Turing machine M_A to semidecides A. We label the strings in Σ^* as $s_1, s_2, \cdots$ in increasing length.

$M_A =$ on input $"M"$:
1. for $i = 2023, 2024, \cdots$
2.   for $s = s_1, s_2, \cdots, s_i$
3.   if s is a palindrome
4.   run M on s for i steps
5.   if M halts on at least 2023 palindromes
6.   halt

This completes the proof.

Q4. Bonus

(a) Firstly, we proved that if $B \leq A$ then B is recursive. In class we have proved $A_{CFG} = \{"G""w" : G \text{ is a CFG that generates } w\}$ is recursive. There is a CGF G_A generates A, so $A \leq A_{CFG}$ by $f(w) = "G_A""w"$, thus A is recursive, then B is recursive.

(b) Secondly, we proved that if B is recursive, then $B \leq A$. We can construct a reduction function f from B to A as follows. Here, B is recursive, so $f(w)$ is computable.

$$\text{If } w \in B, f(w) = 01 \in A,$$
$$\text{If } w \notin B, f(w) = 00 \notin A.$$

Then $w \in B$ iff $f(w) \in A$. Thus, $B \leq A$.