1. Consider the language

EQ = { M # N | M and N are Turing machines with $\mathscr{L}(M) = \mathscr{L}(N)$ }.

(a) Prove that EQ is not recursive.

$$HP \leq EQ$$

$$M \# x \mapsto N_1 \# N_2$$

N_1, on input v_1, accept and halt.

N_2, on input v_2,

Simulate M on x

If the simulation halts, accept and halt.

M halts on x

$$\mathcal{L}(N_1) = \mathcal{L}(N_2) = \Sigma^*$$

M does not halt on x

$$\mathcal{L}(N_1) = \Sigma^*, \quad \mathcal{L}(N_2) \neq \Sigma^*$$

$$= \emptyset$$

(b) Is EQ recursively enumerable?

$\overline{HP} \leq EQ$

$M \# x \longmapsto N_1 \# N_2$

N_1, on input v_1, accepts (and halts).

N_2, on input v_2, does:

1. simulate M on x for $|v_2|$ steps.
2. If the simulation halts, reject and halt
   else accept and halt.

M does not halt on x

$\Rightarrow \mathcal{L}(N_1) = \mathcal{L}(N_2) = \Sigma^*$

M halts on x (in s steps)

$\Rightarrow \mathcal{L}(N_1) = \Sigma^*$, $\mathcal{L}(N_2) \neq \Sigma^*$

$= \{ v_2 \mid |v_2| < s \}$

2. Let L be a recursively enumerable language which is not recursive. What type of language is each of the following?

$\bar{L}$ is not RE. (L could be HP, MP, ...)

(a) $A = \{ 0w \mid w \in L \}$

A is RE but not recursive.

$\hookrightarrow L = \mathcal{L}(T)$

$L \leq A$

$w \longmapsto 0w$

On input x,
check whether the first symbol of x is 0.
If not, reject and halt.
Otherwise $x = 0w$
Simulate T on w
Accept if T accepts / Reject if T rejects.

(b) $B = \{ 0w \mid w \notin L \} = \{ 0w \mid w \in \overline{L} \}$

Not RE

$\overline{L} \leq B$

$w \longmapsto 0w$

(c) $C = A \cup B = \{ 0w \mid w \in L \text{ or } w \notin L \} = 0\Sigma^*$ is regular

$\Rightarrow$ CF

$\Rightarrow$ CS

$\Rightarrow$ recursive

(d) $D = \{ 0w \mid w \in L \} \cup \{ 1w \mid w \notin L \}$

$L \leq D \Rightarrow D$ is not recursive

$w \longmapsto 0w$

$\overline{L} \leq D \Rightarrow D$ is not RE.

$w \longmapsto 1w$

3. Let L be a language over Σ, R a recursive language over Σ, and L' = L – R.
Prove/Disprove:

$R = \mathcal{L}(T)$ for a total TM T.

(a) If L is RE, then L' is RE. True

$L = \mathcal{L}(M)$ for some TM M.

A recognizer for L'

On input x

Run T on x

If T accepts, reject and halt.

Run M on x and do whatever M does

(b) If L' is RE, then L is RE. ✗

$L' = L - R$

Take $R = \Sigma^*$

$L' = \emptyset$

$L = \overline{HP}$

(c) If L is RE but not recursive, then L' is RE but not recursive. ✗

$R = \Sigma^*$

$L = HP$

(d) If L' is RE but not recursive, then L is RE but not recursive.

$L' \leq L$

$w \longmapsto w$

$w \in R, L$

$w \notin L'$

but $w \in L$

$L' = L - R.$

L rec

$\overline{R}$ rec

$L' = L \cap \overline{R}$

$\rightarrow$ rec

False

L is non-RE

L' is RE but not Recursive

$A \rightarrow$ RE not R

$$L = \{0w \mid w \in A\} \cup \{1w \mid w \notin A\}$$

$$R = 1\Sigma^*$$

$$L' = \{0w \mid w \in A\}.$$

4. Let CFL be the set of all Turing machines whose languages are context-free.

(a) Using Rice's theorem, prove that neither CFL nor its complement is RE.

Part 2

$$\underset{\text{CF}}{\emptyset} \subseteq \underset{\text{not CF}}{\{a^n b^n c^n \mid n \geq 0\}} \subseteq \underset{\text{CF}}{\{a, b, c\}^*}$$

being CF is not monotone

being non-CF is not monotone

(b) Without using Rice's theorem, prove that neither CFL nor its complement is RE.

$\overline{HP} \leq CFL$ $\longrightarrow$ $M \# x \longmapsto N$

M does not halt on x $\Rightarrow \mathcal{L}(N)$ is CF $\emptyset$

M halts on x $\Rightarrow \mathcal{L}(N)$ is not CF. $\{ww \mid w \in \Sigma^*\}$

$\overline{HP} \leq \overline{CFL}$

$M \# x \longmapsto N$

M does not halt $\Rightarrow \mathcal{L}(N)$ is not CF

M halts $\Rightarrow \mathcal{L}(N)$ is CF.

- Make a $\underbrace{\text{limited time}}_{|y| \text{ steps}}$ simulation of M on x
- Does not halt $\longrightarrow$ Check if $y = ww \longrightarrow A / R$ (in s steps)
- halts $\longrightarrow$ R

$\{ww \mid |w| < \frac{s}{2}\}$ is finite.

5. Let M be a Turing machine.

Rice's theorem

(a) Decidable or not: Whether $\mathscr{L}(M) = \mathscr{L}(M)^R$?

(b) Semidecidable or not: Whether $\mathscr{L}(\mathrm{M}) = \mathscr{L}(\mathrm{M})^{\mathrm{R}}$?

6. (a) State and prove Rice's theorem (Part 1) for pairs of RE languages.

$$RE^2 = \{ (L_1, L_2) \mid L_1, L_2 \in RE \}$$

Property of RE^2 $\quad P : RE^2 \longrightarrow \{T, F\}$

$(L_1, L_2) \rightarrow$ specified by $M_1 \# M_2$ s.t.

$\mathcal{L}(M_1) = L_1$ and $\mathcal{L}(M_2) = L_2$.

Trivial $\quad P(L_1, L_2) = T \quad \forall L_1, L_2$

$\quad P(L_1, L_2) = F \quad \forall L_1, L_2$

Any non-trivial property of RE^2 is undecidable.

<u>TPT</u> $\quad \Pi = \{ M_1 \# M_2 \mid P(\mathcal{L}(M_1), \mathcal{L}(M_2)) = T \}$

is not recursive.

$$HP \leq \Pi$$

$$M \# x \longmapsto M_1 \# M_2$$

$$P(\emptyset, \emptyset) = F$$

$$P(A, B) = T$$

$$\mathcal{L}(N_1) \qquad \mathcal{L}(N_2)$$

M halts on x,

$$P(\mathcal{L}(M_1), \mathcal{L}(M_2)) = T$$

M does not halt on x,

$$P(\mathcal{L}(M_1), \mathcal{L}(M_2)) = F$$

$$\mathcal{L}(M_1) = \mathcal{L}(N_1) = A$$

$$\mathcal{L}(M_2) = \mathcal{L}(N_2) = B$$

$$\mathcal{L}(M_1) = \emptyset$$

$$\mathcal{L}(M_2) = \emptyset.$$

(b) Prove that { M # N | $\mathscr{L}$(M) = $\mathscr{L}$(N) } is not recursive.

EQ =

in a non-trivial property.

7. Let M be a Turing machine. Prove that it is decidable whether M, on a given input w, moves left at least 2021 times.

Simulate M on w for

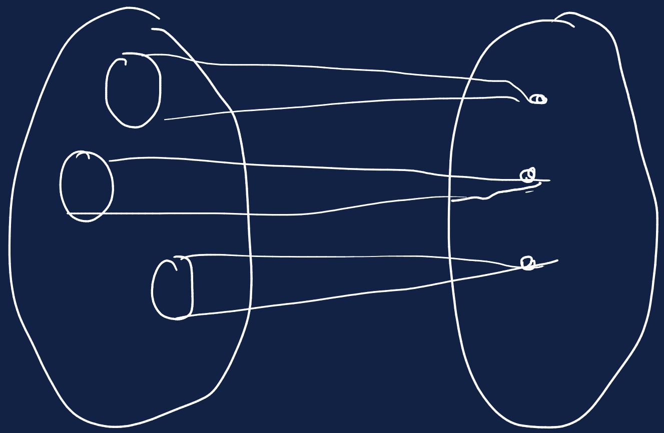
$|Q| + |w| + 2021$ steps.

If 2021 left movements found, accept

Reject.

8. Let A be a language over Σ, and B a language over Λ. Suppose that A $\leqslant_m$ B under a reduction map $\Sigma^* \to \Lambda^*$ which is onto (surjective). Prove that B $\leqslant_m$ A.

$f : \Sigma^* \to \Lambda^*$



$g : \Lambda^* \to \Sigma^*$

$w \longmapsto$ a preimage of w under f.

Given w, a <u>total</u> TM can compute $g(w)$ and halt.

9. Prove that the following problems about DFA D, D_1, D_2 are decidable.

(a) Whether $\mathscr{L}(D) = \varnothing$.
(b) Whether $\mathscr{L}(D) = \Sigma^*$.
(c) Whether $\mathscr{L}(D_1) = \mathscr{L}(D_2)$.

(a) Check whether any accept state is reachable
(b) ______________ non-accept state is reachable

Graph traversal

(c) Let $A = \mathscr{L}(D_1)$ and $B = \mathscr{L}(D_2)$

Build a DFA for $A \triangle B = (A \cap \bar{B}) \cup (\bar{A} \cap B)$

$A = B$ if and only if $A \triangle B = \varnothing$