Final exam

10:30–12:30 pm, Monday, 11 January 2021 (2021/01/11)

Name : 李宥霆
Student ID : B07902048

Note:

1. Write down clearly your name and student ID in the space above.
2. This is a closed book exam.
3. No discussion is allowed.
4. There are SIX questions altogether.
5. Write your solution for each question in the space provided.
6. You can freely use any result proved/stated in the class including pumping lemma (for both regular languages and CFL).

Some notations/definitions

- For a TM $\mathcal{M}$, $L(\mathcal{M})$ denotes the language that consists of all the words accepted by $\mathcal{M}$. That is, $L(\mathcal{M}) = \{w \mid \mathcal{M} \text{ accepts } w\}$.
- $\mathsf{HALT} := \{\lfloor\mathcal{M}\rfloor\$w \mid \mathcal{M} \text{ accepts } w\}$.
- $\mathsf{HALT}_0 := \{\lfloor\mathcal{M}\rfloor\$w \mid \mathcal{M} \text{ accepts } \lfloor\mathcal{M}\rfloor\}$.
- $\mathsf{HALT}'_0 := \{\lfloor\mathcal{M}\rfloor\$w \mid \mathcal{M} \text{ does not accept } \lfloor\mathcal{M}\rfloor\}$.
- $\mathsf{HALT}_{k,\ell} := \{\lfloor\mathcal{M}\rfloor\$w \mid \mathcal{M} \text{ has } \leqslant k \text{ states, } w \text{ has length } \leqslant \ell \text{ and } \mathcal{M} \text{ accepts } w\}$.
  In this definition the states in a TM can be arbitrary 0-1 strings. The point here is that if $\lfloor\mathcal{M}\rfloor\$w \in \mathsf{HALT}_{k,\ell}$, then $\mathcal{M}$ has less than or equal to k states.

Question 1 (2 points). Consider the following regex e over the alphabet $\Sigma = \{a, b\}$.

$$e \ := \ \left(b^*(abb^*)^*(\emptyset^* \cup a)\right) \ \cup \ \left(\Sigma^* aa \Sigma^* bb \Sigma^*\right) \ \cup \ \left(\Sigma^* bb \Sigma^* aa \Sigma^*\right)$$

Determine which of the following words are in $L(e)$. You don't need to prove your answer. Just answer "yes" or "no."

(a) aaa.

(b) $bbaa$.

(c) $abab$.

(d) $baba$.

Solution for question 1.

(a) No

(b) Yes

(c) Yes

(d) Yes

Question 2 (2 points). Consider the following language L over the alphabet $\Sigma = \{a, b\}$.

$$L := \{a^n b^n a^k b^k \mid n, k \geqslant 0\} \cup \{a^k b^n a^n b^k \mid n, k \geqslant 0\}$$

Is L CFL? If your answer is "yes," give the CFG. If your answer is "no," present the proof.

Solution for question 2.

Yes,

$S \to X \mid Y$

$X \to AA$

$A \to aAb \mid \varepsilon$

$Y \to aYb \mid B$

$B \to bBa \mid \varepsilon$

Question 3 (2 points). Consider the following Turing machine A that works as follows.

INPUT: $\lfloor \mathcal{M} \rfloor \$ w$.

- Construct a TM $\mathcal{K}_{\mathcal{M},w}$ that works as follows.

  INPUT: $u \in \Sigma^*$.
  - Run $\mathcal{M}$ on w.
  - If $\mathcal{M}$ accepts w, do the following:
    * Check if $u = 111$.
    * If it is, ACCEPT. (Here ACCEPT is for $\mathcal{K}_{\mathcal{M},w}$ to accept u.)
    * Otherwise, REJECT. (Here REJECT is for $\mathcal{K}_{\mathcal{M},w}$ to reject u.)
  - If $\mathcal{M}$ rejects w, REJECT. (Here REJECT is for $\mathcal{K}_{\mathcal{M},w}$ to reject u.)
- Output $\lfloor \mathcal{K}_{\mathcal{M},w} \rfloor$.

Answer each of the following questions

(a) If $\mathcal{M}$ accepts w, what is the language $L(\mathcal{K}_{\mathcal{M},w})$?

(b) If $\mathcal{M}$ rejects w, what is the language $L(\mathcal{K}_{\mathcal{M},w})$?

(c) If $\mathcal{M}$ does not halt on w, what is the language $L(\mathcal{K}_{\mathcal{M},w})$?

(d) Define the language $L_{111} := \{\lfloor \mathcal{M} \rfloor \mid \mathcal{M} \text{ accepts the word } 111\}$. Is the following true?

$$\lfloor \mathcal{M} \rfloor \$ w \in \mathsf{HALT} \quad \text{if and only if} \quad \lfloor \mathcal{K}_{\mathcal{M},w} \rfloor \in L_{111}$$

Justify your answer (in a few sentences).

Solution for question 3.

(a) $L(K_{M,w}) = \{111\}$

(b) $L(K_{M,w}) = \emptyset$

(c) $L(K_{M,w}) = \emptyset$

(d)

True

if $\lfloor M \rfloor \$ w \in HALT \rightarrow L(K_{M,w}) = \{111\}$ (a)

if $\lfloor M \rfloor \$ w \notin HALT \rightarrow L(K_{M,w}) = \emptyset$ (b) + (c)

#

Question 4 (2 points). For an integer $n \geqslant 1$, define the language L_n as follows.

$$L_n := \{\lfloor \mathcal{M} \rfloor \mid \mathcal{M} \text{ accepts at most } n \text{ words}\}$$

Equivalently,

$$L_n := \{\lfloor \mathcal{M} \rfloor \mid |L(\mathcal{M})| \leqslant n\}$$

Prove that for every $n \geqslant 1$, the language L_n is undecidable.

Solution for question 4. Prove by contradiction, (Turing Reduction)

On Input $\lfloor M \rfloor \$ w$, $L_n = \overline{L_\infty}$

Construct a TM M_1 works as follow:

On Input u,

- Run M on w
- if M accept w, ACCEPT
- if M reject w, REJECT

M_d is the decider of L_∞

Run M_d on Input $\lfloor M_1 \rfloor$, if M_d return true, return FALSE

else, return TRUE

if M accept $w \Rightarrow L(M_1) = \Sigma^*$, M_d return True, function return FALSE

if M not accept $w \Rightarrow L(M_1) = \emptyset$, M_d return False, function return True #

(accept no word)

Question 5 (2 points). For integers $k, \ell \geqslant 1$, define language $Z_{k,\ell}$ as follows.

$$Z_{k,\ell} \;:=\; \{\lfloor \mathcal{M} \rfloor \$ w \mid \text{the length of } \lfloor \mathcal{M} \rfloor \text{ is } \leqslant k, \text{ the length of } w \text{ is } \leqslant \ell \text{ and } \mathcal{M} \text{ accepts } w\}$$

Note that $Z_{k,\ell}$ is different from $\mathsf{HALT}_{k,\ell}$ defined on the front page.

- Prove that for every $k, \ell \geqslant 1$, the language $Z_{k,\ell}$ is *regular*. DFA
- In the class we show that $\mathsf{HALT}_{k,\ell}$ is decidable. Is $\mathsf{HALT}_{k,\ell}$ regular? Justify your answer.

See the front page for the definition of $\mathsf{HALT}_{k,\ell}$.

Solution for question 5.

- Since $|\lfloor M \rfloor| \le k$, $|w| \le \ell$, then $|\lfloor M \rfloor \$ w| \le k + \ell + 1$

  And since types of M is finite (because of finite length),
  types of w is finite (because of finite length),
  There is finite $\lfloor M \rfloor \$ w$ we can find.

  $\because$ As above, we can enumerate all possible $\lfloor M_i \rfloor \$ w_j$ $(i, j \in N)$,
  and each $|\lfloor M_i \rfloor \$ w_j| \le k + \ell + 1$, thus we union all possible answer,
  the language is regular #

- Yes, Since there are finite states in TM M, the construction of type of M are finite, and type of w is finite either, so we can enumerate all possible $\lfloor M_i \rfloor \$ w_j$ that is in $\mathsf{HALT}_{k,\ell}$, thus the union of all possible $\lfloor M_i \rfloor \$ w_j$ is the regular language #

Question 6 (to qualify for A+). Prove that the following problem CFL-Complement is undecidable.

CFL-Complement	
Input:	A CFG $\mathcal{G} = \langle \Sigma, V, R, S \rangle$.
Task:	Output True, if $\Sigma^* - L(\mathcal{G})$ is a CFL. Otherwise, output False.

Note: For this question, you can assume (without proof) that for every Turing machine $\mathcal{M}$, for every word w, the length of the run of $\mathcal{M}$ on w is bigger than or equal to 2. That is, $\mathcal{M}$ does *not* accept/reject any word in 0 or 1 step.

You can use (without proof) the fact that $L_\infty := \{\lfloor \mathcal{M} \rfloor \mid \mathcal{M}$ accepts infinitely many words$\}$ is undecidable.

Solution for question 6.

Below is a prove of $L_\infty \leq_m \overline{\text{CFL-Complement}}$.

M not accept → CFL → Lemma 1

M accept: $\lfloor M \rfloor \in L_\infty$ → not CFL → Lemma 2-1

M accept: $\lfloor M \rfloor \notin L_\infty$ → CFL → Lemma 2-2

Lemma 1: words that is the run of M that does not accept, the set of the words is a CFL (computational history)

Lemma 2-1: words that is the run of M that accept, and $\lfloor M \rfloor \in L_\infty$, the set of the words is not a CFL

Lemma 2-2: words that is the run of M that accept, and $\lfloor M \rfloor \notin L_\infty$, the set of the words is a CFL

I will prove these Lemmas later.

Below is a mapping function f that:

On input $\lfloor M \rfloor$,

Output: the CFG that is the set of words which are the runs of M that does not accept.

- If $\lfloor M \rfloor \in L_\infty$, $\Sigma^* - L(f(\lfloor M \rfloor))$, which means the words which are the runs of M that accept, is not a CFL, By Lemma 2-1.
- If $\lfloor M \rfloor \notin L_\infty$, $\Sigma^* - L(f(\lfloor M \rfloor))$, is a CFL, By Lemma 2-2.

Solution for question 6.

For now, we've proved that $L_\infty \le_m \overline{\text{CFL-complement}}$. And Since that, we've also proved that $L_\infty \le_m$ CFL-complement.

☆ Proof of Lemma 1:

Here, we use a similar way that proved the ALL_{CFG} by using computational history that is taught in class.

— To check whether a run of input W is valid, we need to check 3 parts:

- Whether the start configuration is valid: we can simply construct a PDA that check about the start configuration, If it is valid, return False, else return True.
- Whether the end configuration is valid: If the configuration is end in a accepting state, return False, else return True.
- Whether the transition between the config is valid: Here we inverse the config that is even number in the seqence. i.e. $C_1 \# C_2^R \# C_3 \# C_4^R \ldots\ldots$, then we can simply use an PDA to check whether it is valid. If valid, return False, else, return True.

To here, we can see that the Lemma 1 has been proved because we can convert the PDA to CFG.

☆ Proof of Lemma 2-2:

Since $\langle M \rangle \notin L_\infty$, there are finite accepting words that can be generated by TM M, thus we can contruct a CFG that generates all words in M #

☆ Proof of Lemma 2-1:

Because L_∞ is undecidable, set of words is not CFL