INT201-Ass2 Ruiyang.Wu sID:2257475

Question 1

In each case below, find a CFG generating the given language.

- The set of odd-length strings in {a, b}* with middle symbol a. (10 Marks)
- The set of even-length strings in {a, b}* with the two middle symbols equal. (10 Marks)
- The set of odd-length strings in {a, b}* whose first, middle, and last symbols are all the same. (10 Marks)

(30 Marks Total)

(a): CFG $G_a = (V, \Sigma, R, S)$

Variables $V = \{S\}$

Terminals $\Sigma = \{a, b\}$

Rules R: $S \to aSa \mid aSb \mid bSa \mid bSb \mid a$

Start Symbol S: S

(b) CFG $G_b = (V, \Sigma, R, S)$

Variables $V = \{S\}$

Terminals $\Sigma = \{a, b\}$

Rules R: $S \to aSa \mid aSb \mid bSa \mid bSb \mid aa \mid bb$

Start Symbol S: S

(c) CFG $G_c = (V, \Sigma, R, S)$

Variables $V = \{S, X, Y\}$

Terminals $\Sigma = \{a, b\}$

Rules R: $S \to aXa \mid bYb \mid a \mid b$ (if length is 1).

$X \to aXa \mid aXb \mid bXa \mid bXb \mid a$

$Y \to aYa \mid aYb \mid bYa \mid bYb \mid b$

Start Symbol S: S

Question 2

Closure properties of Turing-recognizable languages.

Are Turing-recognizable languages closed under Kleene star? Prove it or provide a counter example. (10 Marks)

Are Turing-recognizable languages closed under complement? Prove it or provide a counter example. (10 Marks)

Are Turing-decidable languages closed under complement? Prove it or provide a counter example. (10 Marks)

(30 Marks Total)

(a) Proof: let L be a Turning-recognizable language.

Kleene Star $L^* = \{w_1 w_2 \cdots w_n \mid w_i \in L, n \geq 0\}$

if $n=0$, $L^* = \epsilon$, it can be accept.

else, construct a turning machine M', it can split input string w into all possible substrings decompositions of w

i.e, $w \xrightarrow{M'} w^{(1)} = w_1^{(1)} w_2^{(1)} \cdots w_{n_1}^{(1)}$

$w \xrightarrow{M'} w^{(2)} = w_1^{(2)} w_2^{(2)} \cdots w_{n_2}^{(2)}$

$\vdots$

$w \xrightarrow{M'} w^{(m)} = w_1^{(m)} w_2^{(m)} \cdots w_{n_m}^{(m)}$

if M accept all substrings in any decomposition of w, then M' accept w.

since M' can recognize any string in L^*, so L^* is turning-recognizable.

$\therefore$ Turing-recognizable languages are closed under Kleene star.

(b) Turing-recognizable languages are NOT closed under complement

Counter example:

let L be a Turning-recognizable language.

$\therefore$ Exist a TM, for input $w \in L$, it must halt

$H = \{\langle M, w\rangle \mid M \text{ halts on input } w\}$

In complement of the upper Halting Problem:

$H^c = \{\langle M, w\rangle \mid M \text{ does not halt on input } w\}$.

Due to the undecidability of the halting problem, we cannot detect whether M does not halt on input w.

so H^c is not Turing-recognizable, which is a counter example of the topic.

(c) Proof: let L be a Turing-decidable language.

$\therefore$ Exist a TM, will halt and accept for $w \in L$,

will halt and reject for $w \notin L$.

In complement case: construct a TM M' decide L^c:

$$\begin{cases} \text{If } M \text{ accept } w, M' \text{ reject } w. \\ \text{If } M \text{ reject } w, M' \text{ accept } w. \end{cases}$$

so for $\forall w \in L^c$, M' will halt.

$\therefore$ Turing-decidable languages are closed under complement.

Question 3

Show the language $A_{CFG} = \{\langle G\rangle | G \text{ is a CFG that generates letter } a\}$ is decidable.

(20 Marks Total)

Proof: The decision problem is to determine if G only generates letter "a". ① we can convert it into equivalent grammar in CNF. ② All the derivations can be listed in $(2n-1)$ steps, where $|n| = w$, except if $n=0$, then instead list all derivations with 1 step. ③ if any diverations in the corresponding CNF generate "a", accept; if not, reject.

Since the production rules is finite in CNF, we always know whether it reject or accept. So A_{CFG} is decidable.

Question 4

Show the language T = $\{\langle M\rangle \mid$ M is a TM that accepts ww whenever it accept $w\}$ is undecidable.

(20 Marks Total)

*By Rice's Theorem, any nontrivial property of the language recognized by a Turing machine is undecidable. To use Rice's Theorem, we must show that the property is non-trivial: The property must depend only on the language L(M), not on the specific details of M; there exists at least one Turing machine whose language satisfies the property and at least one Turing machine whose language does not satisfy it.

Proof: To show language $T = \{<M> | M$ is a TM that accepts ww whenever it accept $w\}$, we can proof the language have non-trivial properties and According to Rice's Theorem, it is undecidable.

① Property in question: $P(L(M)) =$ If $w \in L(M), \Rightarrow ww \in L(M)$

$\therefore$ It's a property of language $L(M)$, Not the structure of TM, so we can apply Rice's Theorem.

② Not empty: Consider a TM M_1 such that $L(M_1) = \emptyset$,
(M_1 accepts no strings.)
It already satisfies the property: "if $w \in L(M_1)$ then $ww \in L(M_1)$"
So there exist at least 1 TM whose language satisfies the property.

③ Sub set: Consider a TM M_2, such that $L(M_2) = \{w\}$
(M_2 only accept string w)
$\because w \in L(M_2)$, $ww \notin L(M_2)$.
$\therefore L(M_2)$ does not satisfy the property.
So there exist at least 1 TM whose language Not satisfies the property.

So by Rice's theorem, the property $P(L(M))$ is non-trival,
Therefore, the language T is undecidable.

Q.E.D.