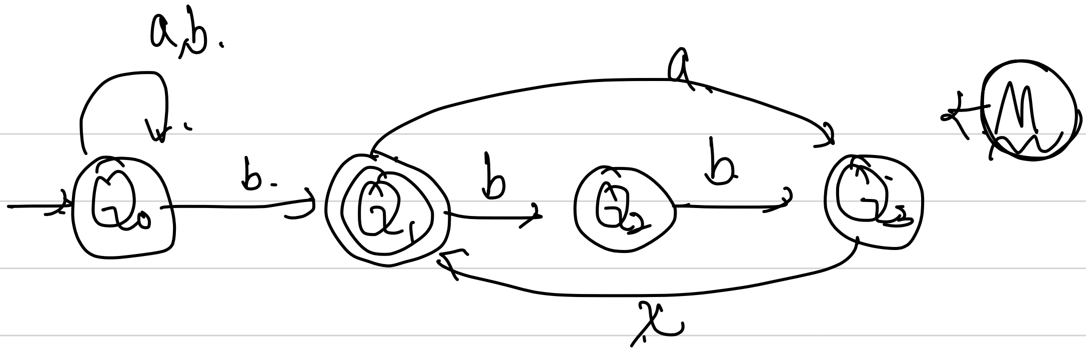
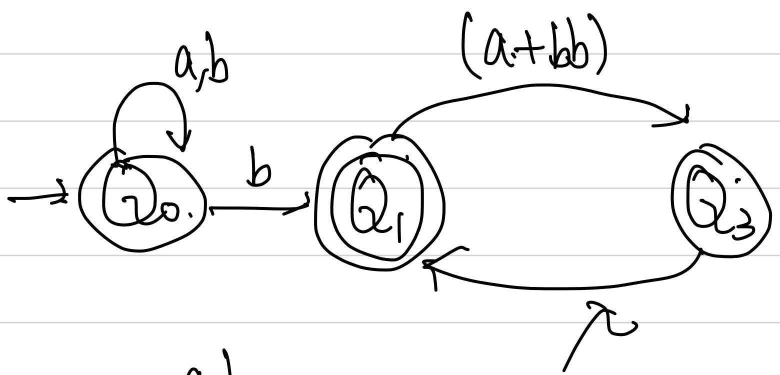
#1.

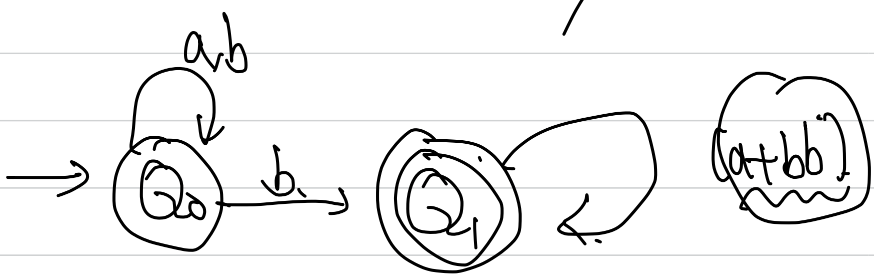


(1) M에 의해 accept 되는 길이가 3인 string을 모두 찾으시오

aab, abb, bbb, bab, aba, bba, baa — 7개.

(2) L(M)을 regular expression으로 표현하시오.





$$\boxed{(a+b)^* b (a+bb)^*}$$

(3). M에 대응되는 Right-linear grammar의 production.

$$Q_0 \to aQ_0 \mid bQ_0 \mid bQ_1 \qquad \boxed{Q_1 \to \lambda}$$

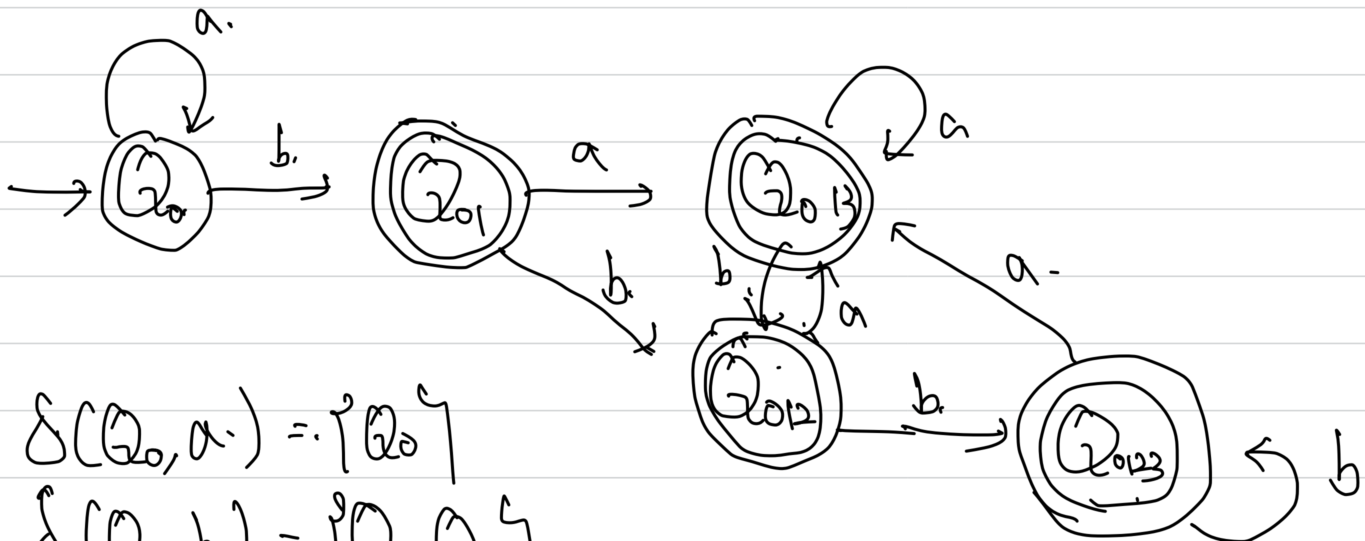
final state 이면 이거 꼭 적어 주어야 한다

$$Q_1 \to bQ_2 \mid aQ_3$$

$$Q_2 \to bQ_3$$

$$Q_3 \to Q_1$$

(4) M과 equivalent 한 DFA로 transition graph를 나타내시오.



$\delta(Q_0, a) = \{Q_0\}$

$\delta(Q_0, b) = \{Q_0, Q_1\}$

$\delta(Q_{01}, a) = \delta(Q_0, a) \cup \delta(Q_1, a) = \{Q_0, Q_3, Q_1\}$

$\delta(Q_{013}, a) = \{Q_0, Q_1, Q_3\}$

$\delta(Q_{01}, b) = \delta(Q_0, b) \cup \delta(Q_1, b) = \{Q_0, Q_1, Q_2\}$

$\delta(Q_{012}, b) = \{Q_0, Q_1, Q_2, Q_3\}$

#2. Grammar $G = (\{S\}, \{a,b\}, S, P)$, $P: S \to aSbS \mid bSaS \mid \lambda$

(1) G에 의해 정의되는 길이가 4인 모든 string을 찾으시오.

$S \Rightarrow aSbS \Rightarrow abSaSbS \Rightarrow abab$

$aSbS \Rightarrow abbSaS \Rightarrow abba$

$aSbS \Rightarrow aaSbSbS \Rightarrow aabb$

$aSbS \Rightarrow aSbaSbS \Rightarrow \cancel{abab}$

$S \Rightarrow bSaS \Rightarrow baSbSaS \Rightarrow baba$

$bSaS \Rightarrow bSaaSbS \Rightarrow baab$

$bSaS \Rightarrow bbSaSaS \Rightarrow bbaa$

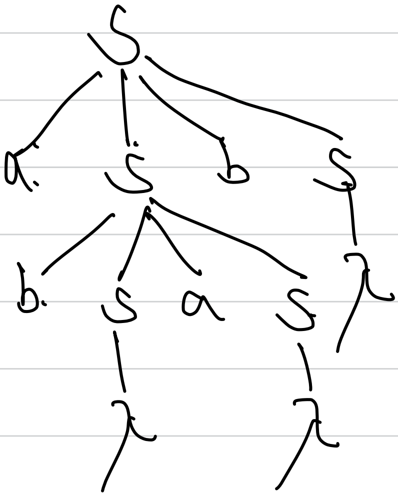
$bSaS \Rightarrow bSabSaS \Rightarrow \cancel{baba}$

6개

(2) G에 의해 정의되는 language L(G)를 집합으로 (조건제시법) 표현하시오.

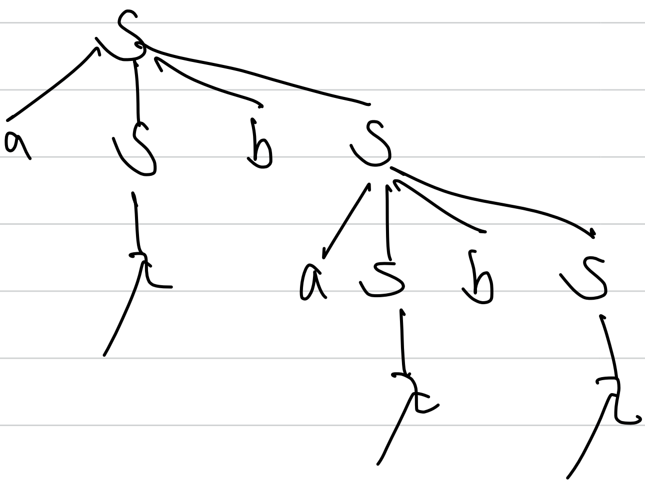
$$L(G) = \{ w \in \{a,b\}^* \mid n_a(w) = n_b(w) \}.$$

(3). 스트링 $\boxed{abab}$ 에 대한 parsing은 가능한가? 가능하면 derivation tree를 그리시오



(4) G는 ambiguous 한가? 이에 대한 답과 이유를 쓰시오

abab 에 대한 derivation tree가 2가지 존재



(5). L은 inherently ambiguous 한가? 이에 대한 답과 이유를 쓰시오.

"S-grammar 를 고려하라,"

$L(G) = \{ w \in \{a,b\}^* \mid n_a(w) = n_b(w) \}$.

abab. 개수 같아야 함

$$Z \to aAZ \mid bBZ \mid \lambda$$
$$A \to b \mid aAA$$
$$B \to a \mid bBB$$

aaAA

S-grammar 가 존재하므로.

unambiguous 하다.

∴ L 은 inherently ambiguous 하지 않다.

#3. 다음 중 regular가 아닌 language를 하나 찾아 pumping lemma를 이용하여 증명하세요.

① $L = \{ vwv : v, w \in \{a,b\}^*, |v| = 2 \}$. → regular

② $L = \{ w_1 c w_2 : w_1, w_2 \in \{a,b\}^*, w_1 \neq w_2 \}$?

③ $L = \{ w \in \{a,b\}^* : n_a(w) + 3n_b(w) \text{ is even} \}$?

pumping lemma.

④ $L = \{ uv : u \in L((ab)^*), v = (L((ab)^*))^R \}$ → regular.

L is regular 라고 가정

② → $\forall m$. choose $\exists$ $w = a^m c a^{m+1}$ $(|w| \ge m)$.

Consider $\forall$ decomposition of xyz $(|xy| \le m, |y| \ge 1)$

$y = a^k \ (1 \le k \le m)$.

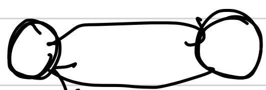
$a^{m-k} c \, a^{m+1} \notin L$.

$\therefore$ L is not regular

#4. 다음 명제의 참, 거짓을 ~~판단하고~~ 그 이유를 간단히 쓰시오.

(1) 모든 regular language는 unambiguous 하다

DFA



L로 가는 경로가 유일하다

$\left(\begin{array}{l} A \rightarrow x. \\ A \in V \quad x \in (V \cup T)^* \end{array} \right)$

$\therefore$ unambiguous 하다.

L을 이러면

(2) L이 context free language 이면 L^* 도 context-free language 이다.

CFG

$G = (V, \Sigma, S, P)$

시작 vertex.

$(S_0 \rightarrow SS_0)$

$S_0 \Rightarrow SS_0 \Rightarrow SSS_0$

L^*이 되는 데 이는 context-free가 된다.

(3) L_1과 L_2가 nonregular 이면 $L_1 \cup L_2$도 nonregular 이다.

반례를 생각해보자.

$L_1 = \{a^n b^m \mid n = m\}$ ← not-regular

$L_2 = \{a^n b^m \mid n \neq m\}$ - not-regular

$$L_1 \cup L_2 = \{a^n b^m \mid n \geq 0, m \geq 0\}$$

$= a^* b^* \rightarrow$ reg expr 이

존재.

False

(4) 임의의 regular language L_1, L_2에 대해 L_1이 L_2의 진부분집합 인지를 판단하는 알고리즘이 존재한다.

True

$L_1 \subset L_2$ and $L_2 - L_1 \neq \emptyset$ 임을 판단 해야 함.

① $L_1 \subseteq L_2$. $L_1 \cup L_2 = L_2$ 임을 판단.

모두 regular language 이므로 알고리즘 존재

② $L_2 - L_1 \neq \emptyset$ 인 경우도. $L_2 - L_1 = L_2 \cap L_1^c$ 은 regular 이므로 알고리즘이 존재 한다.

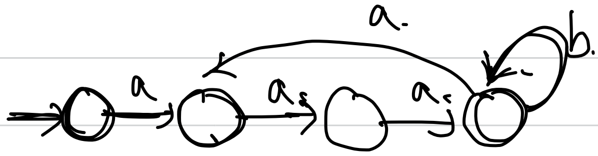
(5). $L = \{a^n b^m \mid n,m \ge 1, (nm) \bmod 3 = 0\}$은 nonregular 하다.

False

$(nm) = 3 \cdot 6 \cdot 9 \cdots$

$n = 3k,\ m = 1$ $\quad a^3 b^1 \quad a^6 b^1$

3 1
3 2
3 3



이 경우 nfa 존재

⇒ regular

(6). $L((r_1 r_2)^*) = L(r_1^* r_2^*)$

a. b.

false.

"$(ab)^*$ 와 a^*b^* 는 다르다,

(ab)(ab) $\quad$ aabbb ...

(7). $L = \{a^n b^m \mid n = m+1, m \ge 0\}$을 나타내는 reg expression이 존재한다.

regular 아님

$L(a^{m+1} b^m)$. $\quad$ a. reg expression.

$L \cup \{a\} = \{a^n b^n \mid n \ge 1\}$ not regular,

L 은 regular language 가 아니다

→ reg expression은 존재하지 않는다.

$L(a^{m+1} b^m) = L$,

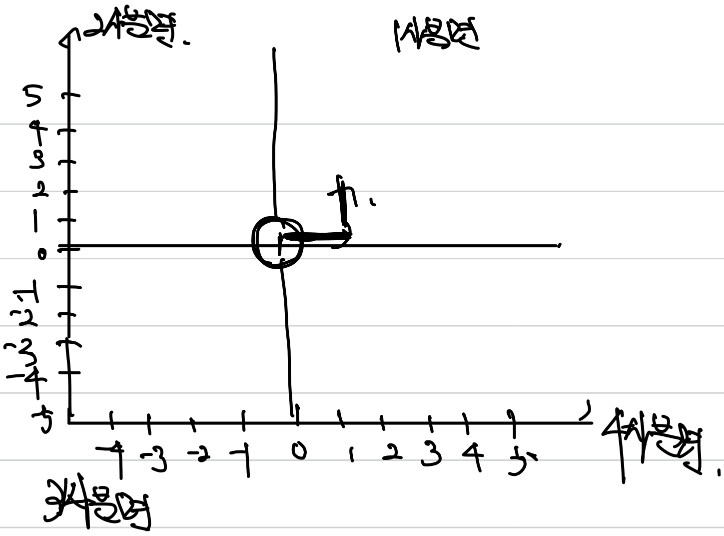
$L_1 = \{b\}$

$L \cdot L_1 = \{a^n b^n \mid n \ge 1\}$

$\{a^n b^n \mid n \ge 0\} - \{\lambda\}$.

regular language X

#.



두 동전 → (상/하/좌/우) → 랜덤하게 움직인다.

1사분면 → accept → 그 외의 경우 → reject.

FA를 찾을 수 있는지. 가능하면 FA를 찾고, 불가능하면 이유를 적으시오.

ⓐ ⓑ ⓒ ⓓ
상 하 좌 우
$n(a) > n(b)$ $n(c) < n(d)$ 인 경우.

$L = \{ w \in \{a, b, c, d\}^* \mid n(a) > n(b) \text{ and } n(c) < n(d) \}$.

$\forall m \geq 0$. choose $w = a^{m+1} b^m c^m d^{m+1}$ $(|w| \geq m)$

consider all possible decomposition of $w = xyz$. $(|xy| \leq m, |y| \geq 1)$

and y has a form. a^k $(1 \leq k \leq m)$

$w_0 = xz = (a^{m+1-k} b^m c^m d^{m+1}) \notin L$.

$m+1-k \leq m$ ∴ pumping lemma is false

∴ L is not Regular → FA (X)

$A \to x \cdot (A \in V, x \in (V \cup T)^*)$

(b) "Context Free Grammar를 찾을 수 있는가? 가능하다면 이를 찾고, 불가능하다면 그 이유를 쓰시오."

완전 흔히 받은자.

"a나 d로 끝나야 한다"

$$P: \left\{ \begin{array}{l} S \to aSb \mid cSd \mid SS \mid A \mid D \\ A \to aA \mid \lambda \\ D \to dD \mid \lambda \end{array} \right.$$

bSa | dSc

$S \Rightarrow aSb \Rightarrow aaSbb$

$\Rightarrow aacSdbb$

A

만들 수 없음

$G = (\{S, A, D\}, \{a, b, c, d\}, S, P)$