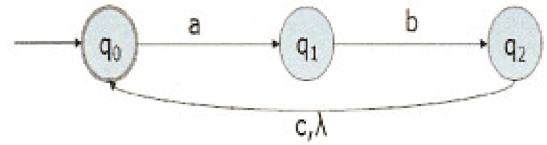
Automata 2009-1학기 중간고사 2009.04.20

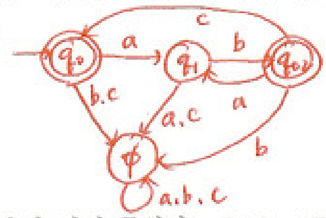
학번 이름

1. 오른쪽 transition graph로 나타나는 NFA에 대해 다음에 답하시오 (20점).
(1) 이 NFA에 의해 accept되는 길이가 5이하인 string을 모두 찾으시오.
(2) (1)의 결과로부터 유추하여 이 NFA에 대응되는 regular expression을 찾으시오.
(3) NFA를 DFA로 바꾸는 알고리즘을 사용하여 대응되는 DFA를 찾으시오.
(4) FA -> Right Linear Grammar로 변환하는 알고리즘을 이용하여 대응되는 Right Linear Grammar를 찾으시오.



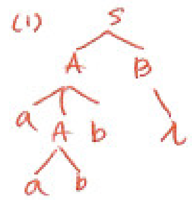
(1) ab, abc, abab, ababc, abcab, λ
(2) $r=(ab(\lambda+c))^*$ 또는 $r=(ab+abc)^*$
(4) $G=(\{q_0,q_1,q_2\}, \{a,b,c\}, q_0, P)$,
$q_0 \rightarrow aq_1 \mid \lambda$ $q_1 \rightarrow bq_2$ $q_2 \rightarrow q_0 \mid cq_0$

(3)



2. Grammar $G=(\{A,B,C,D,E\}, \{a,b\}, S, P)$가 다음과 같이 주었다: P: $S\rightarrow AB \mid aC$, $A\rightarrow aAb \mid ab$, $B\rightarrow bB \mid \lambda$, $D\rightarrow B$, $E\rightarrow DBa$
(1) $w = aabb$의 derivation tree를 그리시오. (3점)
(2) λ-production, unit-production, useless production을 가지지 않는 equivalent grammar 찾으시오. (7점)
(3) (2)에서 찾은 grammar를 Chomsky Normal Form으로 바꾸시오. (5점)
(4) CYK algorithm을 이용하여 $w = aaab$의 membership을 판단하시오. (5점)

(1)



(2) $S\rightarrow AB \mid aAb \mid ab$, $A\rightarrow aAb \mid ab$, $B\rightarrow bB \mid b$
(3) $S\rightarrow AB \mid CE \mid CD$, $A\rightarrow CE \mid CD$, $B\rightarrow DB \mid b$
$C\rightarrow a$, $D\rightarrow d$
(4) $V_{11}=V_{22}=V_{33}=\{C\}$, $V_{44}=\{D\}$
$V_{12}=V_{23}=\{\}$, $V_{34}=\{S,A\}$, $V_{13}=\{\}$, $V_{24}=\{\}$, $V_{14}=\{\}$ --> reject!

3. 핸드폰에서 많이 사용되는 문자입력 방식을 생각한다. 자음 {ㄱ,ㄴ,ㄷ,...,ㅎ,ㄲ,ㄸ,ㅃ,ㅆ,ㅉ}은 하나의 심볼 c로 나타내고, 모음을 만드는 기본 요소 {ㅡ}, {ㅣ}, {·}는 각각 심볼 h, v, d로 나타낸다. 이 기본 심볼들을 조합하여 다음 한글의 생성규칙에 따라 하나의 한글이 만들어진다고 가정한다.
이 규칙에 의해 만들어지는 한글문장을 원소로 가지는 language를 L이라고 정의할 때, 다음 질문에 답하시오.(10점)

[한글의 생성규칙]	
[1] 자음 + 수평단모음(ㅗ,ㅛ,ㅜ,ㅠ,ㅡ) or 수직단모음(ㅏ,ㅑ,ㅓ,ㅕ,ㅣ)	(예: 고, 갸)
[2] 자음 + 수직복모음(ㅐ,ㅒ,ㅔ,ㅖ)	(예: 개, 데)
[3] 자음 + 수평단모음(ㅗ,ㅛ,ㅜ,ㅠ,ㅡ) + 수직단모음(ㅏ,ㅑ,ㅓ,ㅕ,ㅣ) or 수직복모음(ㅐ,ㅒ,ㅔ,ㅖ)	(예: 과, 돼, 웨)
[4] [1] or [2] or [3] + 자음	(예: 공, 갖, 궠)

(1) L을 생성하는 context-free grammar를 정의하시오. (힌트: 심볼의 집합 {c, h, v, d}를 알파벳으로 가지고, 생성규칙에서 사용된 기본 요소들(자음, 수평단모음, 수직복모음 등)에 대응되는 variable들을 정의하여 사용)
(2) L은 regular 인가? 판단하고 그 이유를 쓰시오.

(1) G=({S, H(받침없는 한글문자), K(받침있는 한글문자), A(수평단모음), B(수직단모음), D(수직복모음)}, {c,h,v,d}, S,P)
$S\rightarrow HS \mid KS$, $H\rightarrow cA \mid cB \mid cD \mid cAB \mid cAD$ $K\rightarrow Hc$,
$A\rightarrow dh \mid ddh \mid hd \mid hdd \mid d$ $B\rightarrow vd \mid vdd \mid dv \mid ddv \mid v$ $D\rightarrow vdv \mid vddv \mid dvv \mid ddvv$
(2) regular expression으로 표현 가능

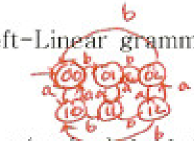
4. 참/거짓을 판단하고 이유를 쓰시오 (35점).
(1) 모든 finite language는 regular이고, unambiguous하다.
T Finite language는 finite state를 이용하여 FA를 만들 수 있으므로 regular.
DFA는 하나의 string에 대해 하나의 path만 존재하므로 unambiguous.
(2) L_1과 L_2가 regular가 아니면, $L_1 \cup L_2$도 regular가 아니다.
F 반례: $L_1=\{a^nb^n \mid n \geq 0\}$, $L_2=\{a^nb^l \mid n \neq l\}$, $L_1 \cup L_2=\{a^*b^*\}$
(3) regular expression r_1, r_2로 주어지는 어떤 Language $L(r_1), L(r_2)$에 대해,
$L(r_1{}^*) = L(r_1 + r_2)$가 성립하는지 판단하는 알고리즘이 존재한다.
T $L(r_1), L(r_2)$는 regular expression으로 표현되므로 regular. 따라서 $L(r_1{}^*), L(r_1+r_2)$도 regular.
두 regular laguage가 같은지 여부를 판단하는 알고리즘은 존재.

(4) 모든 regular grammar는 unambiguous하다.

F 반례: S → aS | aA, A → aA|a 로 정의되는 grammar는 regular이지만 ambiguous grammar임
(aaa에 대한 서로 다른 derivation tree가 존재)

(5) $L=\{w \in a,b^* |\ n_a(w) \bmod 2 = 0,\ n_b(w) \bmod 3 = 0\}$일 때, L을 생성하는 Left-Linear grammar가 존재한다.

T L을 accept하는 DFA를 만들 수 있으며 (그림참조), 따라서 L은 regular.
그러므로 L을 생성하는 Left-Linear Grammar가 존재.



(6) 임의의 homomorphism h와 regular language L_1, L_2에 대해, $h(L_1)/h(L_2) = h(L_1/L_2)$가 성립한다.

F 반례: L_1={aab}, L_2={a} h(a)=a, h(b)=a 로 정의하면,
$h(L_1)/h(L_2)$ = {aaa}/{a}={aa}, $h(L_1/L_2)$=h({aab}/{a})=$h(\varnothing)=\varnothing$

(7) 모든 Linear Language는 Regular이다.

F 반례: S → aSb|ab에 의해 생성되는 language는 Linear. 그러나 이 language L={$a^n b^n |\ n>0$}는 regular가 아님.

5. 다음 물음에 답하시오. (15점)

(1) $L=\{w \in a,b^* | n_a(w) \ge n_b(w)\}$가 regular가 아님을 pumping lemma를 이용하여 증명하시오. (8점)

Assume L is regular.

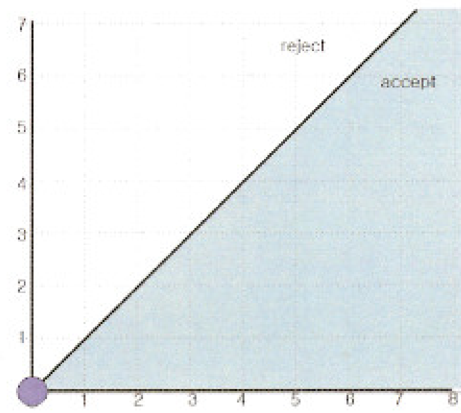
Then for any given m>0, we choose $w = a^m b^m$

For any possible decomposition $w = xyz\,(|xy| \le m, |y| \ge 1)$, y has the form, $y = a^k (1 \le k \le m)$

For given $y = a^k$, we set $i=0$. Then $w_0 = a^{m-k} b^m \notin L$

This is contradiction to pumping lemma. Therefore, L is not regular.

(2) 오른쪽 그림과 같이 2차원 격자로 나타나는 공간의 원점에 로봇을 두고, 한 번에 오른쪽이나 위쪽으로 움직인다. n번 움직인 후에 로봇의 위치가 그림에서 색칠한 부분(직선포함)에 있는 경우 accept, 그렇지 않은 경우는 reject하는 DFA를 만드는 것은 가능한가? 판단하고 그 이유를 쓰시오. (7점)



불가능하다.

오른쪽으로 움직인 것을 a, 위쪽으로 움직인 것을 b라고 하면,

n번 움직인 후에 로봇의 위치를 length가 n인 a,b로 이루어진 string w로 나타낸다.

accept되는 string의 집합을 나타내면

$L=\{w \in a,b^* | n_a(w) \ge n_b(w)\}$가 되어 위 (1)과 같아짐.

(1)에서 L은 regular가 아님이 증명되었으므로, DFA는 존재하지 않음.