

1. NPDA $M = (\{q_0, q_1, q_2\}, \{a, b\}, \{a, b, z\}, \delta, q_0, z, \{q_2\})$ 이 정의되었다. (20점)
 $\delta(q_0, a, z) = \{(q_1, az)\}$, $\delta(q_2, z) = \{(q_2, z)\}$, $\delta(q_1, b, a) = \{(q_1, b)\}$, $\delta(q_1, b, b) = \{(q_1, b)\}$, $\delta(q_1, a, b) = \{(q_2, \lambda)\}$
 (1) M 에 의해 accept되는 길이 4이하의 string을 모두 찾으시오.

a. aba, abba

- (2) M 에 의해 accept되는 language $L(M)$ 을 집합으로 나타내시오.

$$L(M) = \{ab^n a \mid n \geq 1\}$$

- (3) 같은 language를 accept하는 DPDA를 만드는 것은 가능한가? 판단하고 이유를 쓰시오.

가능함.

$$\delta(q_0, a, z) = (q_1, az) \text{로 해}$$

$$\delta(q_1, b, a) = (q_2, b), \delta(q_1, b, b) = (q_2, b) \text{로}$$

$$\delta(q_2, a, b) = (q_1, \lambda) \text{로 각각 바꿀때 } q_1 \text{을 final로 함.}$$

* 기타 유사한 방법으로 DPDA를 만들어 주면 정답처리.

- (4) M 의 Final state가 $F = \{q_1, q_2\}$ 로 바뀌었을 때, $L(M)$ 을 집합으로 나타내시오.

$$L(M) = \{ab^n \mid n \geq 0\} \cup \{ab^m a \mid m \geq 1\}$$

2. 두 개의 stack을 가지는 NPDA를 생각한다. 다음 물음에 답하시오. (10점)

- (1) Formal definition을 쓰시오.

$$M = (Q, \Sigma, \Gamma_1, \Gamma_2, \delta, q_0, z_1, z_2, F)$$

$$\delta: Q \times \Sigma \times \Gamma_1 \times \Gamma_2 \rightarrow 2^{Q \times \Gamma_1^* \times \Gamma_2^*}$$

- (2) Turing machine과 power를 비교하고 그 판단의 이유를 쓰시오.

power는 같다.

NPDA의 두 stack이 하나의 tape로 동작될수 있도록 재구성하면 TM은 simulation 할수 있음.

즉, stack 1에서 pop 된것이 stack 2에 push 되어서 tape에서의 head 움직임을 표현.

본반

학번

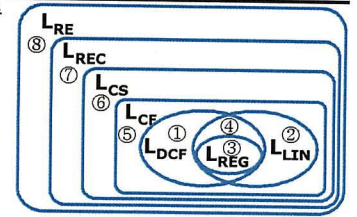
이름

3. 다음 각 Language가 오른쪽 그림의 영역 ①부터 ⑧ 중 어디에 속하는지 말하고, 그 이유를 간단히 쓰시오. (20점)
 (1) $L = \{a^{n^2} : n \geq 0\}$

답 ⑥.

CF가 아닌 이유는 pumping Lemma로 증명.

CS인 이유는 LBA를 만들수 있음 (만드는 방법 간단히 설명).



- (2) $L = \{w \in \{a, b, c\}^* \mid n_a(w) + n_b(w) = n_c(w)\}$

답 ①.

LIN이 아닌 이유는 pumping lemma로 증명

DCF인 이유는 DPDA를 만들수 있음을 간단히 설명.

4. 다음 명제의 참, 거짓을 판단하고 간단히 증명하시오. (20점)

(1) L_1, L_2 가 recursive이면 $L_1 - L_2$ 도 recursive이다.

True.

$$L_1 - L_2 = L_1 \cap L_2^c$$

↑
recursive.

L_1 과 L_2^c 가 둘다 membership algorithm을 가지므로 두 개가 동시에 accept 하는 것만 accept 하는 TM을 만들수 있음.

(2) context sensitive language는 reverse연산에 닫혀있지 않다.

False.

CSL은 LBA가 풀수.

L^R 을 위한 TM은 L 을 위한 LBA에서 입력순만 거꾸로 하면 되므로 여전히 bound된 TM이 만들어질.

(3) 모든 unrestricted grammar의 집합은 countable하다.

True.

Unrestricted grammar는 TM과 1대1 대응
모든 TM의 집합은 countable.

(4) L_1 이 context-free이고, L_2 가 regular일 때, L_1 과 L_2 에 동시에 속하는 string이 있는지 판단하는 알고리즘이 존재한다.

True.

$L_1 \cap L_2$ 는 context free. (regular intersection에 닫혀있으므로)
context free language는 emptiness를 판단하는 알고리즘이 존재.

[유의사항]

- 개별적 질문은 받지 않음. 문제 해석에 의문이 있는 경우에는 자신이 이해한 내용을 기술한 후 풀이를 작성하면 채점 시에 참고로 할 예정.
- 교과서 본문에 증명되어 있는 사실들은 별도의 증명과정 없이 풀이에서 인용 가능함.

5. binary notation(이진법)으로 표현된 양의 정수 x 를 입력받아 어떤 함수 $f(x)$ 를 계산하는 Turing Machine M_f 가 주어졌다. (20점)

$$M_f = (\{q_0, q_1, q_2\}, \{0, 1\}, \{0, 1, \square\}, \delta, q_0, \square, \{q_2\})$$

$$\delta(q_0, 0) = (q_0, 0, R), \delta(q_0, 1) = (q_0, 1, R), \delta(q_0, \square) = (q_1, 0, L),$$

$$\delta(q_1, 0) = (q_1, 0, L), \delta(q_1, 1) = (q_1, 1, L), \delta(q_1, \square) = (q_2, \square, R)$$

(1) 아래와 같은 초기 상황이 주어졌을 때, M_f 의 transition 과정을 기술하시오.

$$q_0 110 \vdash (q_0 10 \vdash (q_0 0 \vdash (10 q_0 \square \vdash (1 q_1 00 \vdash (q_1 100 \vdash (q_1 100 \vdash (q_1 \square 1100 \vdash (q_2 1100$$

(2) M_f 이 계산하는 함수 $f(x)$ 는 무엇인지, 그 수학적식을 쓰시오.

$$f(x) = 2x$$

(3) 위에서 정의된 M_f 의 transition function의 요소들 중 하나만 수정하여 $g(x) = 2x+1$ 을 계산하는 Turing Machine M_g 을 만들고자 한다. 어떤 것을 어떻게 바꾸면 좋을지 쓰시오.

$$\delta(q_0, \square) = (q_1, 1, L) \text{ 호 바꾸기.}$$

(4) 위에서 얻어진 두 turing machine M_f 와 M_g 의 state들을 각각 $\{q_{f0}, q_{f1}, q_{f2}\}$ 와 $\{q_{g0}, q_{g1}, q_{g2}\}$ 로 기호를 바꾸어 표시하였다. 이외에 아래와 같은 transition들을 추가하여 정의되는 새로운 turing machine M_h 가 계산하는 함수 $h(x)$ 의 정의를 쓰시오.

$$M_h = (\{p_0, p_f, p_g\} \cup \{q_{f0}, q_{f1}, q_{f2}\} \cup \{q_{g0}, q_{g1}, q_{g2}\}, \{0, 1\}, \{0, 1, \square\}, \delta, p_0, \square, \{q_{f2}, q_{g2}\})$$

$$\delta(p_0, 0) = (p_f, 0, L), \delta(p_0, 1) = (p_g, 1, L), \delta(p_f, \square) = (q_{f0}, \square, R), \delta(p_g, \square) = (q_{g0}, \square, R)$$

$$h(x) = \begin{cases} 2x & \text{19 첫 바가 0.} \\ 2x+1 & \text{20 첫 바가 1.} \end{cases}$$

6. Universal Turing Machine에 대해 간단히 설명하시오. (10점)

현재의 범용 컴퓨터와 같이 program이 가능한 TM.

standard TM을 변형시켜 3개의 tape을 가진 TM을 만들고,

각 tape에 ① 특정 TM에 대한 description code, ② 특정 TM에 주어진 입력, ③ 특정 TM의 현재 state 등. 을 기록하여 처리.