

(정의 6.1) 비결정적(Nondeterministic) Pushdown Automata(PDA) P 는 요소 일곱 개로 이루어져있다. 즉 $P = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$ 로 쓰고

- (1) 상태어휘 Q 은 상태(state)의 유한 집합(state vocabulary)이다.
- (2) 입력어휘 Σ 은 입력문자의 유한 집합(input vocabulary)이다.
- (3) 스택어휘 Γ 은 스택문자의 유한 집합(stack vocabulary)이다.
- (4) δ 는 상태변화함수(state transition function)로서 상태집합 Q 와 입력문자(Σ) 혹은 빈 문자열(ϵ), 스택문자(Γ)를 정의역으로, 상태 Q 와 스택문자열의 집합¹⁾을 함수의 치역으로 가진다. 즉

$$\delta: Q \times (\Sigma \cup \{\epsilon\}) \times \Gamma \rightarrow 2^{Q \times \Gamma^*}$$

$(p, \beta) \in \delta(q, a, X)$ 이면, 현 상태 $q \in Q$ 에서 입력문자 $a^2) \in \Sigma \cup \{\epsilon\}$ 와, 스택 top 문자 $X \in \Gamma$ 를 보고, 상태가 $p \in Q$ 로 바뀌고, 스택 top 문자 X 를 pop한 다음에 스택 문자열 $\beta \in \Gamma^*$ 를 push하여³⁾ 스택 top이 β 로 바뀐다.

- (5) $q_0 \in Q$ 는 처음상태(initial state)라고 불리는 특별한(distinguished) 상태이다.
- (6) $Z_0 \in \Gamma$ 는 처음스택내용(initial stack contents)이라고 불리는 특별한(distinguished) 스택문자이다.
- (7) $F \subseteq Q$ 는 끝나는 상태(final state)라고 불리는 특별한 상태들의 집합이다.

(정의 6.2) PDA $P = (Q, \Sigma, \Gamma, \delta, q_0, \gamma_0, F)$ 에서 상황(instantaneous description, configuration)을 (상태, 입력문자열, 스택문자열)의 세 순서쌍⁴⁾으로 정의한다.

$$(q, x, \gamma) \in Q \times \Sigma^* \times \Gamma^*$$

(정의 6.3) PDA $P = (Q, \Sigma, \Gamma, \delta, q_0, \gamma_0, F)$ 에서 상태변화함수 δ 에 따른 상황변화 \vdash_P 를 상황에 관한 이진관계⁵⁾로 정의한다. $\vdash_P \subseteq (Q \times \Sigma^* \times \Gamma^*) \times (Q \times \Sigma^* \times \Gamma^*)$.

$$(q, ax, X\gamma) \vdash_P (p, x, \beta\gamma), \text{ if } (p, \beta) \in \delta(q, a, X).$$

$$(q, x, X\gamma) \vdash_P (p, x, \beta\gamma), \text{ if } (p, \beta) \in \delta(q, \epsilon, X).$$

PDA P 가 명백하면, \vdash_P 에서 P 를 생략하고 \vdash 로 쓴다.

(정의 6.4) 상황변화 \vdash 의 반복횟수(recursive definition) $\vdash^n, n \geq 0$.

$$\vdash^0 \stackrel{\text{B}}{\text{def}} id_{Q \times \Sigma^* \times \Gamma^*}.$$

$$\vdash^n \stackrel{\text{R}}{\text{def}} \vdash \circ \vdash^{n-1}.$$

(정의 6.5) 상황변화 \vdash 의 반복합(closure) \vdash^\dagger 와 \vdash^* .

- 1) 상태와 스택문자열의 집합을 치역으로 허용하고 ϵ -move도 허용하므로 nondeterministic하다.
- 2) ϵ -NFA와 같이 입력 문자 a 가 Σ 의 원소이면 다음 입력문자가 a 의 오른쪽 문자가 되고, a 가 ϵ 이면 다음 입력문자가 변하지 않는다.
- 3) 스택 문자열 $\beta = X_1 X_2 \dots X_n$ 일 때, 오른쪽 끝 문자 X_n 부터 차례로 거꾸로 push하여 β 의 가장 왼쪽 문자 X_1 이 스택 top에 들어난다.
- 4) ordered triple
- 5) Binary relation on $Q \times \Sigma^* \times \Gamma^*$ (상황).

$$\vdash^\dagger \stackrel{\text{def}}{=} \bigcup_{i \in \mathbb{N}_1} \vdash^i = \vdash^1 \cup \vdash^2 \cup \vdash^3 \cup \dots$$

$$\vdash^* \stackrel{\text{def}}{=} \bigcup_{i \in \mathbb{N}_0} \vdash^i = \vdash^0 \cup \vdash^1 \cup \vdash^2 \cup \dots$$

(정의 6.6) PDA $P = (Q, \Sigma, \Gamma, \delta, q_0, \gamma_0, F)$ 가 정의하는 언어, $L(P)$.

$$L(P) = \{x \in \Sigma^* \mid (q_0, x, Z_0) \vdash_P^* (f, \epsilon, \gamma), f \in F\} \quad \text{Final state accept}$$

$$N(P) = \{x \in \Sigma^* \mid (q_0, x, Z_0) \vdash_P^* (q, \epsilon, \epsilon), q \in Q\}. \quad \text{Null stack accept}$$

Final state accept와 Null stack accept를 모두 수용하기 위하여, (5) 처음상태 $q_0 \in Q$ 와 (6) 처음스택내용 $\gamma_0 \in \Gamma^*$ 를 처음상황 $\iota = (q_0, x, \gamma_0) \in Q \times \Sigma^* \times \Gamma^*$ 으로 바꾸고, (7) 끝나는 상태 $F \subseteq Q$ 을 끝나는 상황집합 $\Phi \subseteq Q \times \Sigma^* \times \Gamma^*$ 로 다시정의하면, Final state acceptance는 $\Phi_F = \{(f, \epsilon, \alpha) \mid \alpha \in \Gamma^*\}$ 로 정의하고, Null stack acceptance는 $\Phi_N = \{(q, \epsilon, \epsilon) \mid q \in Q\}$ 로 처리하면 되므로, 7-tuple PDA $P = (Q, \Sigma, \Gamma, \delta, q_0, \gamma_0, F)$ 를 6-tuple $P = (Q, \Sigma, \Gamma, \delta, \iota, \Phi)$ 로 다시 정의할 수 있다.

(정의 6.7) 변화된 PDA $P = (Q, \Sigma, \Gamma, \delta, \iota, \Phi)$ 에서

- (1) 상태어휘 Q , (2) 입력어휘 Σ , (3) 스택어휘 Γ , (4) 상태변화함수 δ 는 원래 PDA와 같고,
- (5) 처음상황(initial configuration) $\iota \in Q \times \Sigma^* \times \Gamma^*$,
- (6) 끝나는 상황 집합(set of final configurations) $\Phi \subseteq Q \times \Sigma^* \times \Gamma^*$ 이다.

(정의 6.8) PDA $P = (Q, \Sigma, \Gamma, \delta, \iota, \Phi)$ 가 받아들이는 언어(language)는

$$L(P) = \{x \in \Sigma^* \mid \iota \vdash_P^* \phi, \phi \in \Phi\} \text{이다.}$$

7-tuple PDA $P = (Q, \Sigma, \Gamma, \delta, q_0, \gamma_0, F)$ 를 6-tuple PDA $P = (Q, \Sigma, \Gamma, \delta, \iota, \Phi)$ 로 다시 정의하며 초기상황 $\iota = (q_0, x, \gamma_0)$ 마지막상황 집합 $\Phi_{Final\ state} = \{(f, \epsilon, \alpha) \mid f \in F, \alpha \in \Gamma^*\}$ 와 $\Phi_{Null\ stack} = \{(q, \epsilon, \epsilon) \mid q \in Q\}$ 두 가지를 생각하면 두 개의 6-tuple PDA $P_{Final\ state} = (Q, \Sigma, \Gamma, \delta, \iota, \Phi_{Final\ state})$ 와 $P_{Null\ stack} = (Q, \Sigma, \Gamma, \delta, \iota, \Phi_{Null\ stack})$ 을 정의할 수 있고,

$$L(P) = L(P_{Final\ state}) \text{와 } N(P) = L(P_{Null\ stack}) \text{로 통합된다.}$$

(정의 6.10) 확장된 PDA $P = (Q, \Sigma, \Gamma, \Delta, \iota, \Phi)$

- (1) 상태어휘 Q , (2) 입력어휘 Σ , (3) 스택어휘 Γ 은 원래(변화된) PDA와 같고,
- (4) 상황변화규칙 $\Delta \subseteq (Q \times \Sigma^* \times \Gamma^*) \times (Q \times \Sigma^* \times \Gamma^*)$ 으로 확장한다.
 상황변화함수(configuration transition function) δ 대신에 상황변화규칙 Δ 를 써서, 상황 (Q, Σ^*, Γ^*) 에 관한 이진관계(binary relation)로 본다.
 - i) 읽는 스택문자를 하나(Γ)에서 0개 이상(Γ^*)으로 확장한다.
 - ii) 입력문자(Σ)나 ϵ 을 입력문자열(Σ^*)로 확장하지만, 입력 테이프는 읽을 수 만 있으므로(read only) $(q, xy, \alpha) \rightarrow (p, y, \beta)$ 인 형태만 허용한다.

iii) 읽고 쓰는 스택 문자(Γ)는 스택 문자열(Γ^*)로 확장한다.

상황변화규칙 $(q, xy, \alpha) \rightarrow (p, y, \beta)$ 를 7-tuple PDA에서는 상황변화함수 $(p, y, \beta) \in \delta(q, xy, \alpha)$ 로 쓸 수 있다.

(5) 처음상황 ι 과 (6)마지막상황 집합 Φ 도 변화된 PDA와 같다.

(사실) 제한된 PDA로 확장된 PDA를 흉내 낼 수 있다.

확장된 PDA를 다시쓰기(Rewriting) PDA를 다시 정의한다.

(정의 6.11) 다시쓰기 PDA $R = (Q \times \Sigma^* \times \Gamma^*, \Delta, \iota, \Phi)$ 은

(1) 상태어휘 Q , 입력어휘 Σ , 스택어휘 Γ 은 원래 PDA와 같고,

$Q \times \Sigma^* \times \Gamma^*$ 은 상황(configuration)의 집합이다.

(2) $\Delta \subseteq (Q \times \Sigma^* \times \Gamma^*) \times (Q \times \Sigma^* \times \Gamma^*)$ 는 상황변화규칙으로

$(q, xy, \alpha) \rightarrow (p, y, \beta) \in P$ 로 쓰고6),

임의의 입력 문자열 $z \in \Sigma^*$ 와 스택 문자열 $\gamma \in \Gamma^*$ 에 대하여 \rightarrow 를 \Rightarrow 로 다시쓰기 (rewriting)로 확장하여 $(q, xy, \alpha) \rightarrow (p, y, \beta)$ 이면, $(q, xyz, \alpha\gamma) \Rightarrow (p, yz, \beta\gamma)$ 로 다시 쓰고(rewrite), 상황변화규칙 $(q, xy, \alpha) \rightarrow (p, y, \beta)$ 를 강조하고 싶으면

$(q, xyz, \alpha\gamma) \Rightarrow^{(q, xy, \alpha) \rightarrow (p, y, \beta)} (p, yz, \beta\gamma)$ 로 쓴다.

(3) $\iota \in Q \times \Sigma^* \times \Gamma^*$ 는 초기상황이다.

(4) $\Phi \subseteq Q \times \Sigma^* \times \Gamma^*$ 는 최종상황의 집합이다.

(정의 6.12) 다시쓰기 PDA $R = (Q \times \Sigma^* \times \Gamma^*, \Delta, \iota, \Phi)$ 의 언어(language)는

$L(R) = \{x \in \Sigma^* \mid \iota \Rightarrow^\delta \phi \in \Phi, \delta \in \Delta^*\}$ 이고, $\delta \in \Delta^*$ 는 사용된 규칙열이다.

6) 문법과 같은 기술임에 유의하라.