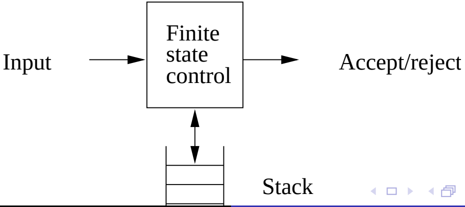
Automi a pila

Automi a pila

[1] Cap. 6.1, 6.2, 6.3.1, 6.4

Un automa a pila (PDA) e' in pratica un ϵ-NFA con una pila.
In una transizione un PDA:

1. Consuma un simbolo di input o nessuno se la transizione e' su ϵ.
2. Va in un nuovo stato (o rimane dove e').
3. Rimpiazza il top della pila con una stringa
(non fa niente, o elimina il top della pila, o mette una stringa in cima alla pila)



Definizione formale di PDA

Un PDA e' una tupla di 7 elementi:

$$P = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F),$$

dove

- Q e' un insieme finito di stati,
- Σ e' un *alfabeto finito di input,*
- Γ e' un *alfabeto finito di pila,*
- $\delta : Q \times (\Sigma \cup \{\epsilon\}) \times \Gamma \to 2^{Q \times \Gamma^*}$ e' la *funzione di transizione,*
- q_0 e' lo *stato iniziale,*
- $Z_0 \in \Gamma$ e' il *simbolo iniziale* per la pila, e
- $F \subseteq Q$ e' l'insieme di *stati di accettazione.*

La funzione δ

Notiamo che funzione δ ha ora tre argomenti:

- uno stato
- un simbolo dell'alfabeto di input o ϵ;
- un simbolo dell'alfabeto di pila (non necessariamente disgiunto; da quello di input).

e restituisce un *insieme* di coppie $(q, \gamma) \in (Q \times \Gamma^*)$.
Quindi, in generale, un PDA e' *non deterministico.*

Esempio

Consideriamo

$$L_{wwr} = \{ww^R : w \in \{0,1\}^*\},$$

generato dalla "grammatica" $P \to 0P0$, $P \to 1P1$, $P \to \epsilon$. Un PDA per L_{wwr} ha tre stati, e funziona come segue:

1. Parte nello stato q_0;
2. Scommette che sta leggendo w. Rimane nello stato q_0, e mette il simbolo di input sulla pila.
3. Scommette che sta nel mezzo di ww^R. Va spontaneamente nello stato q_1.
4. Sta leggendo la testa di w^R. La paragona al top della pila. Se sono uguali, fa un pop della pila, e rimane nello stato q_1. Se non sono uguali, si ferma.
5. Se la pila e' vuota, va nello stato q_2 e accetta la stringa.

Il PDA che riconosce il lingaggio L_{wwr} e' definito dalla 7-tupla

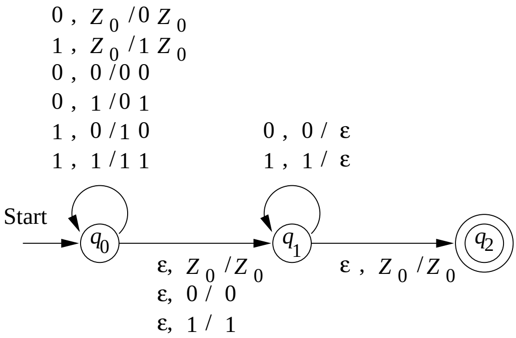
$$P = (\{q_0, q_1, q_2\}, \{0, 1\}, \{0, 1, Z_0\}, \delta, q_0, Z_0, \{q_2\}),$$

dove δ e' data dalla tavola seguente:

	$0, Z_0$	$1, Z_0$	0,0	0,1	1,0	1,1	ϵ, Z_0	$\epsilon, 0$	$\epsilon, 1$
$\rightarrow q_0$	$q_0, 0Z_0$	$q_0, 1Z_0$	$q_0, 00$	$q_0, 01$	$q_0, 10$	$q_0, 11$	q_1, Z_0	$q_1, 0$	$q_1, 1$
q_1			q_1, ϵ			q_1, ϵ	q_2, Z_0		
$\star q_2$									

Per esempio $\delta(q_0, 0, Z_0) = \{(q_0, oZ_0)\ \}$, etc...

Un PDA puo‘ essre rappresentato da un diagramma di transizione in cui le etichette sono della forma $a, X/\gamma$ in cui a e' in simbolo di input, X la cima della pila e γ la stringa che sostituisce X sulla pila. Per esempio il PDA precedente puo' essere rappresentato in questo modo:



Descrizioni istantanee

Un PDA passa da una configurazione ad un'altra configurazione consumando un simbolo di input.
Per ragionare sulle computazioni dei PDA, usiamo delle **descrizioni istantanee** (ID) del PDA. Una ID e' una tripla

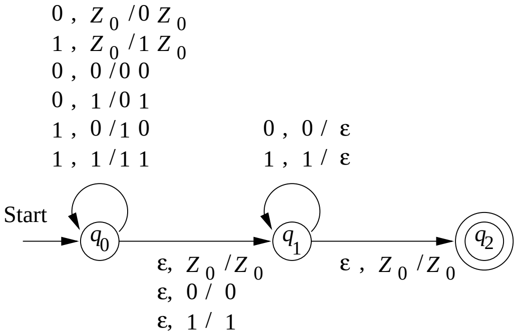
$$(q, w, \gamma)$$

dove q e' lo stato, w l'input rimanente, e γ il contenuto della pila.
Sia $P = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$ un PDA. Allora $\forall w \in \Sigma^*, \beta \in \Gamma^*$:

$$(p, \alpha) \in \delta(q, a, X) \Rightarrow (q, aw, X\beta) \vdash (p, w, \alpha\beta).$$

Definiamo $\overset{*}{\vdash}$ la chiusura riflessiva e transitiva di $\vdash$.

Su input 1111 il PDA



ha le seguenti sequenze di computazioni:

$$
\begin{tikzcd}
(q_0, 1111, Z_0) \arrow[d] \arrow[dr] & & \\
(q_0, 111, 1Z_0) \arrow[d] \arrow[dr] & (q_1, 1111, Z_0) \arrow[r] & (q_2, 1111, Z_0) \\
(q_0, 11, 11Z_0) \arrow[d] \arrow[dr] & (q_1, 111, 1Z_0) \arrow[r] & (q_1, 11, Z_0) \arrow[d] \\
(q_0, 1, 111Z_0) \arrow[d] \arrow[dr] & (q_1, 11, 11Z_0) \arrow[dr] & (q_2, 11, Z_0) \\
(q_0, \varepsilon, 1111Z_0) \arrow[d] & (q_1, 1, 111Z_0) \arrow[d] & (q_1, 1, 1Z_0) \arrow[d] \\
(q_1, \varepsilon, 1111Z_0) & (q_1, \varepsilon, 11Z_0) & (q_1, \varepsilon, Z_0) \arrow[d] \\
 & & (q_2, \varepsilon, Z_0)
\end{tikzcd}
$$

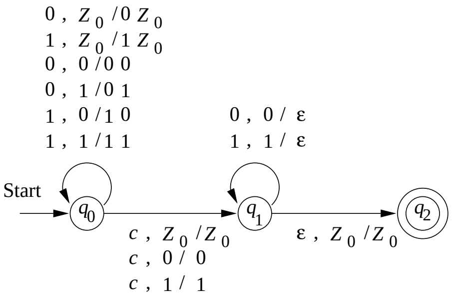
Un PDA $P = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$ e' *deterministico* se e solo se

1. $\delta(q, a, X)$ e' sempre o vuoto o con un solo elemento.
2. Se $\delta(q, a, X)$ non e' vuoto, allora $\delta(q, \epsilon, X)$ deve essere vuoto.

Esempio: Definiamo

$$L_{wcwr} = \{wcw^R : w \in \{0, 1\}^*\}$$

L_{wcwr} e' riconosciuto dal seguente DPDA:



Nota che sostitendo nell'ultima transizione $\epsilon, Z_0/Z_0$ con $\epsilon, Z_0/\epsilon$ si ottiene un DPDA che accetta a pila vuota,

Accettazione per stato finale

Sia $P = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$ un PDA. Il **linguaggio accettato da P per stato finale** e'

$$L(P) = \{w : (q_0, w, Z_0) \overset{*}{\vdash} (q, \epsilon, \alpha),\ q \in F\}.$$

Per esempio il PDA di prima accetta esattamente L_{wwr}. Infatti sia $x \in L_{wwr}$. Allora $x = ww^R$, e la seguente e' una sequenza di computazione legale

$$(q_0, ww^R, Z_0) \overset{*}{\vdash} (q_0, w^R, w^R Z_0) \vdash (q_1, w^R, w^R Z_0) \overset{*}{\vdash}$$

$$(q_1, \epsilon, Z_0) \vdash (q_2, \epsilon, Z_0).$$

Accettazione per pila vuota

Sia $P = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$ un PDA. Il **linguaggio accettato da** P **per pila vuota** e'

$$N(P) = \{w : (q_0, w, Z_0) \overset{*}{\vdash} (q, \epsilon, \epsilon)\}.$$

Nota: q puo' essere uno stato qualunque.
Domanda: come modificare il PDA per le palindromi per accettare lo stesso linguaggio per pila vuota?

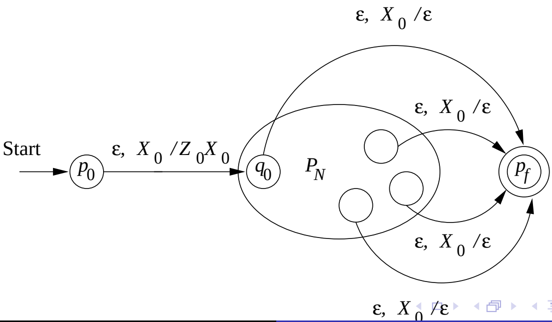
Da pila vuota a stato finale

Proprieta': Se $L = N(P_N)$ per un PDA $P_N = (Q, \Sigma, \Gamma, \delta_N, q_0, Z_0)$, allora $\exists$ PDA P_F, tale che $L = L(P_F)$.

Infatti sia:

$$P_F = (Q \cup \{p_0, p_f\}, \Sigma, \Gamma \cup \{X_0\}, \delta_F, p_0, X_0, \{p_f\})$$

dove $\delta_F(p_0, \epsilon, X_0) = \{(q_0, Z_0X_0)\}$, e per ogni $q \in Q, a \in \Sigma \cup \{\epsilon\}, Y \in \Gamma : \delta_F(q, a, Y) = \delta_N(q, a, Y)$, e inoltre $(p_f, \epsilon) \in \delta_F(q, \epsilon, X_0)$.

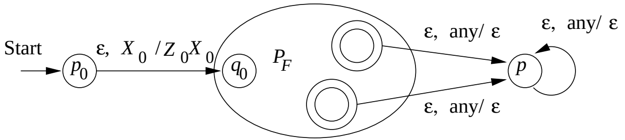


Proprieta': Sia $L = L(P_F)$, per un PDA $P_F = (Q, \Sigma, \Gamma, \delta_F, q_0, Z_0, F)$. Allora $\exists$ PDA P_n, tale che $L = N(P_N)$.

Infatti sia:

$$P_N = (Q \cup \{p_0, p\}, \Sigma, \Gamma \cup \{X_0\}, \delta_N, p_0, X_0)$$

dove $\delta_N(p_0, \epsilon, X_0) = \{(q_0, Z_0 X_0)\}$, $\delta_N(p, \epsilon, Y) = \{(p, \epsilon)\}$, per $Y \in \Gamma \cup \{X_0\}$, e per tutti i $q \in Q$, $a \in \Sigma \cup \{\epsilon\}, Y \in \Gamma : \delta_N(q, a, Y) = \delta_F(q, a, Y)$, e inoltre $\forall q \in F$, and $Y \in \Gamma \cup \{X_0\} : (p, \epsilon) \in \delta_N(q, \epsilon, Y)$.



Equivalenza di PDA e CFG

Un linguaggio e'

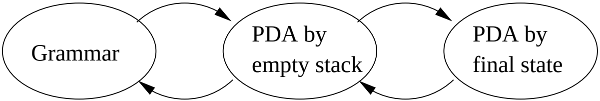
generato da una CFG

se e solo se e'

accettato da un PDA per pila vuota

se e solo se e'

accettato da un PDA per stato finale



Faremo vedere solo come passare da una grammatica ad un automa a pila (sappiamo gia' andare da pila vuota a stato finale e viceversa). Ometteremo la costruzione di una grammatica da un automa a pila.

Data G, costruiamo un PDA che simula $\underset{lm}{\overset{*}{\Rightarrow}}$.

Sia $G = (V, T, Q, S)$ una CFG. Definiamo P_G come

$$(\{q\}, T, V \cup T, \delta, q, S),$$

dove

$$\delta(q, \epsilon, A) = \{(q, \beta) : A \to \beta \ \in \ Q\},$$

per $A \in V$, e

$$\delta(q, a, a) = \{(q, \epsilon)\},$$

per $a \in T$.

Per esempio il PDA corrispondente alla grammatica

$$S \rightarrow aSbS \mid bSaS \mid \epsilon$$

e' definito da:

$$\delta(q, \epsilon, S) = \{(q, aSbS), (q, bSaS), (q, \epsilon)\}$$
$$\delta(q, a, a) = \delta(q, b, b) = \{(q, \epsilon)\}$$

per esempio:

$$(q, ab, S) \vdash (q, ab, aSbS) \vdash (q, b, SbS) \vdash (q, b, bS) \vdash (q, \epsilon, S) \vdash (q, \epsilon, \epsilon)$$

Notiamo che: prendendo la concatenazione della parte consumata dell'input con il contenuto della pila e sostituendo $\vdash$ con $\rightarrow$ si ottiene una derivazione lelftmost (con eventualmente qualche passo ridondante). Infatti:

$$S \rightarrow aSbS \rightarrow^* aSbS \rightarrow abS \rightarrow^* abS \rightarrow ab$$

Proprieta' dei DPDA

- *Non c'e corrispondenza tra PDA e DPDA*. I linguaggi riconosciuti dai DPDA (context ree deterministici) sono un *sottoinsieme* dei linguaggi context free. Per esempio il linguaggio L_{wwr} non puo' essere riconosciuto da nessun DPDA;
- tutti i liguaggi regolari sono accettati dai DPDA (banalmente);
- I DPDA che accettano per pila vuota possono riconoscere solo CFL con la proprieta' del prefisso.
    - Un linguaggio L ha la *proprietá del prefisso* se non esistono due stringhe distinte in L, tali che una é un prefisso dell'altra.
    - Esempio: L_{wcwr} ha la proprietá del prefisso.
    - Esempio: $\{0\}^*$ non ha la proprietá del prefisso.

Notare che se $\$ \notin L$ then $L \cdot \{\$\}$ ha la proprietá del prefisso.