

Parsificazione

(analisi top-down)

Analisi sintattica I

Data una grammatica G l'analizzatore sintattico o parsificatore legge la stringa sorgente e se appartiene al linguaggio $L(G)$ ne produce una derivazione o un albero sintattico, altrimenti si ferma segnalando l'errore.

Due classi importanti di analizzatori

Discendenti o top-down

Ascendenti o bottom-up

Come per l'analisi lessicale, anche per l'analisi sintattica sono stati sviluppati strumenti per la generazione automatica di parsificatori, sia per l'analisi bottom-up (Yacc) sia per quella top-down (Antlr).

Analisi sintattica deterministica

Un analizzatore sintattico *deterministico* leggendo 1 (o più) caratteri in input può *eliminare le ambiguità* e scegliere sempre la strada giusta che porta al riconoscimento della stringa. Ovviamente il modello su cui si rifanno tutti gli analizzatori sintattici (più o meno fedelmente) è l'automa a pila.

Le grammatiche che permettono **parsing predittivo discendente** sono chiamate **LL(k)**, quelle che permettono **parsing predittivo ascendente** sono chiamate **LR(k)**, dove k è il numero di simboli necessari per individuare la produzione senza ambiguità.

La famiglia **LL(k)** contiene tutti e soli i linguaggi che possono essere definiti da una grammatica $LL(k)$ per un valore finito di $k \geq 1$. Non tutti i linguaggi che hanno riconoscitori deterministici sono generabili da grammatiche $LL(k)$, cioè la famiglia dei linguaggi $LL(k)$ è strettamente contenuta nella famiglia dei linguaggi che hanno riconoscitori deterministici.

Verso il parser top-down

Il PDA costruito dalla grammatica: $\{S \rightarrow 0S1 \mid 1A0, A \rightarrow 2A \mid \varepsilon\}$
 riconosce 01201 con questa sequenza di mosse (tra le tante possibili):

$(q, 01201, S) \vdash (q, 01201, 0S1) \vdash (q, 1201, S1) \vdash (q, 1201, 1A01) \vdash$
 $(q, 201, A01) \vdash (q, 201, 2A01) \vdash (q, 01, A01) \vdash (q, 01, 01) \vdash (q, 1, 1) \vdash (q, \varepsilon, \varepsilon)$

	INPUT	PILA	
Togliendo lo stato e lasciando solo l'input e la pila si ottiene la struttura base del parser top-down (iterativo)..	01201	S	<u>Scelte critiche:</u> cosa fare quando abbiamo una variabile sulla pila?.
	01201	0S1	
	1201	S1	
	1201	1A01	Per fare un'analisi efficiente bisogna avere un criterio per fare <i>sempre</i> la mossa giusta.
	201	A01	
	201,	2A01	
	01	A01	
	01	01	
	1	1	
	accetta		

Parser LL(1)

Un parser LL(1) iterativo per una grammatica LL(1) (definita esattamente in seguito) funziona utilizzando:

- la *stringa in input* cui viene aggiunto un mark di fine stringa che denotiamo con \$. L'input viene letto sequenzialmente come negli automi.
- una *pila* i cui elementi possono essere *terminali* o *non terminali* della grammatica
- un *criterio* per decidere la scelta quando c'è una *variabile* in cima alla pila

Durante l'analisi input e pila vengono modificati *esattamente* come l'input e la pila rappresentati nelle configurazioni istantanee di un *automa a pila (non deterministico)* che riconosce la grammatica G costruita nel modo standard.

Nell'algoritmo di parsificazione LL(1) la *decisione* su come espandere i nonterminali in cima alla pila viene presa *in modo deterministico (quando possibile) guardando il primo simbolo dell'input (lookahead)* della stringa ancora da analizzare..

Parsing LL(1) (top-down o discendente)

Consideriamo per esempio la seguente grammatica (LL(1)).
Notare l'uso di "\$" per definire la fine stringa:

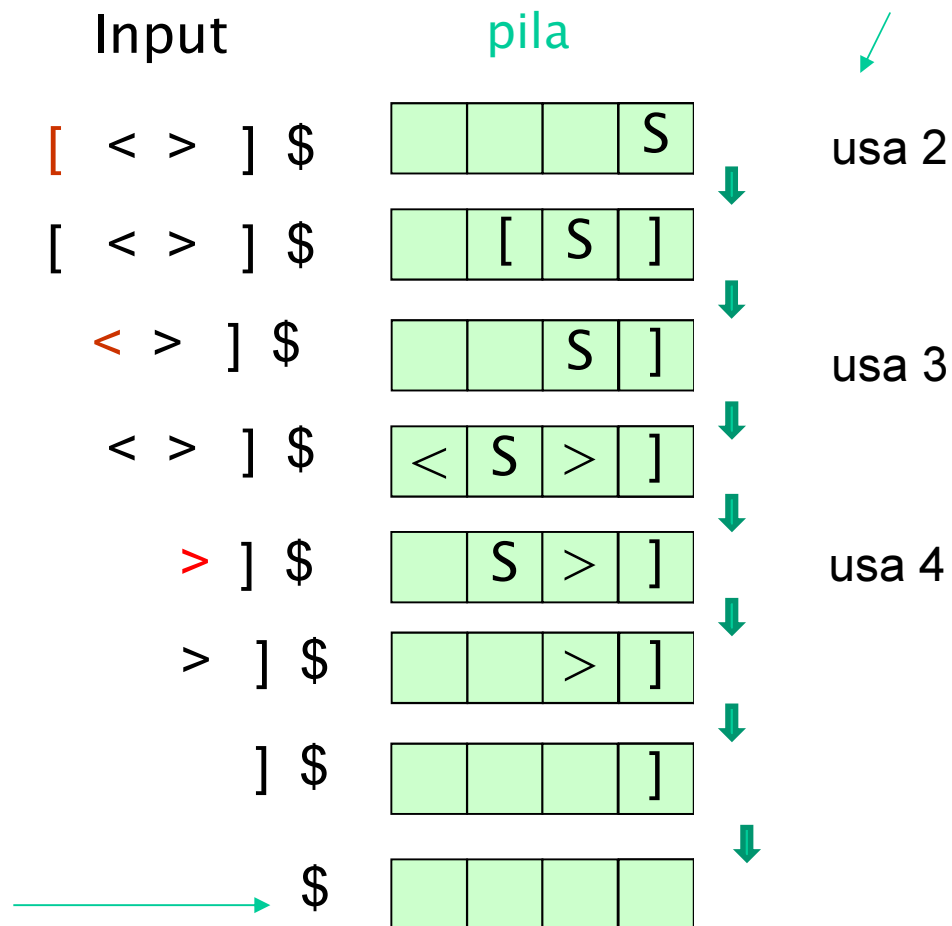
Grammatica

- 1 $S \rightarrow (S)$
- 2 $S \rightarrow [S]$
- 3 $S \rightarrow < S >$
- 4 $S \rightarrow \varepsilon$

Stringa in input: [< >] \$

La decisione su come espandere S
viene presa in funzione dei simboli
in rosso, primi simboli dell'input
ancora da analizzare.

accettazione



FIRST

Data una grammatica $G = \langle V, \Sigma, P, S \rangle$, l'insieme **FIRST** di una stringa α di variabili e terminali, è definito formalmente come:

$$\text{FIRST}(\alpha) = \{a \mid \alpha \rightarrow^* a\beta\} \cup \{\varepsilon \mid \text{se } \alpha \rightarrow^* \varepsilon\}$$

E' l'insieme dei terminali con cui iniziano le stringhe derivabili da α nella grammatica G . **FIRST**(α) (abb. $F(\alpha)$) soddisfa questa definizione (ricorsiva):

$$1. F(\varepsilon) = \{\varepsilon\}$$

$$F(A) = \bigcup_{A \rightarrow \gamma_i \in P} F(\gamma_i)$$

$$2. F(a\beta) = \{a\}$$

$$3. F(A\beta) = \begin{cases} F(A) & \text{se } A \text{ non è annullabile} \\ (F(A) - \{\varepsilon\}) \cup F(\beta) & \text{se } A \text{ è annullabile} \end{cases}$$

N.B. A è annullabile se $A \rightarrow^* \varepsilon$

N.B. $(A \rightarrow \gamma_1 \mid \gamma_2 \mid \dots \mid \gamma_k)$ è l'insieme delle produzioni di A in G

Esempio

$G = \langle \{X, Y, Z\}, \{a, c, d\}, P, Z \rangle$

P: $Z \rightarrow d \mid XYZ$

$Y \rightarrow c \mid \varepsilon$

$X \rightarrow Y \mid a$

$\text{FIRST}(d) = \{d\}$

$\text{FIRST}(XYZ) = \{a, c, d\}$

Infatti $XYZ \rightarrow aYZ$

$XYZ \rightarrow YYZ \rightarrow YZ \rightarrow cZ$

$XYZ \rightarrow YYZ \rightarrow YZ \rightarrow Z \rightarrow d$

$\text{FIRST}\{X\} = \{a, c, \varepsilon\}$

Infatti $X \rightarrow a$

$X \rightarrow Y \rightarrow c$

$X \rightarrow Y \rightarrow \varepsilon$

$\text{FIRST}(Z) = \{a, c, d\}$

Infatti $Z \rightarrow XYZ \rightarrow aYZ$

$Z \rightarrow XYZ \rightarrow YZ \rightarrow cZ$

$Z \rightarrow d$

FOLLOW

Data una grammatica $G = \langle V, \Sigma, P, S \rangle$, l'insieme **FOLLOW** (insieme dei seguiti) di una variabile A è l'insieme dei terminali con cui iniziano le stringhe che seguono A nelle derivazioni della grammatica G (assumendo $\$$ in fine stringa). Formalmente:

$$\text{FOLLOW}(A) = \{a \mid S \rightarrow^* \alpha A a \beta\} \cup \{\$ \mid \text{se } S \rightarrow^* \alpha A\}$$

Notare che $\$$ appartiene sempre al FOLLOW dell'assioma.

FOLLOW(A) (abb. **Fw(A)**) soddisfa la seguente equazione:

$$\begin{aligned} \text{Fw}(A) = & \left[\bigcup_{B \rightarrow \alpha A \beta \in P} (\text{F}(\beta) - \{\epsilon\}) \right] \cup \left[\bigcup_{\substack{B \rightarrow \alpha A \beta \in P \text{ tali che} \\ \beta \text{ annullabile e } B \neq A}} \text{Fw}(B) \right] \cup \\ & \cup \{\$\} \text{ se } A \text{ è lo start symbol di } G \end{aligned}$$

Esempio

$Z \rightarrow d \mid XYZ$

$Y \rightarrow c \mid \varepsilon$

$X \rightarrow Y \mid a$

$\text{FOLLOW}(Y) = \{a, d, c\}$

Infatti: $Z \rightarrow XYZ \rightarrow XYXYZ \rightarrow XY\textcolor{red}{a}YZ$

$Z \rightarrow XYZ \rightarrow XY\textcolor{red}{d}$

$Z \rightarrow XYZ \rightarrow YYZ \rightarrow Y\textcolor{red}{c}Z$

$\text{FOLLOW}(X) = \{c, d, a\}$

Infatti: $Z \rightarrow XYZ \rightarrow X\textcolor{red}{c}Z$

$Z \rightarrow XYZ \rightarrow XZ \rightarrow X\textcolor{red}{d}$

$Z \rightarrow XYZ \rightarrow XYXYZ \rightarrow XXYZ \rightarrow X\textcolor{red}{a}YZ$

$\text{FOLLOW}(Z) = \{\$ \}$ perchè Z è l'assioma.

Insiemi guida

Data una grammatica G , l'**insieme guida** di una produzione della grammatica $A \rightarrow \alpha$ - **Gui ($A \rightarrow \alpha$)** - è l'insieme dei terminali (o \$ se ci si trova a fine parola) con cui *iniziano* le stringhe generabili partendo dalla produzione stessa :

$$\text{Gui}(A \rightarrow \alpha) = \{a \mid S \rightarrow_{\text{lm}}^* wA\beta \rightarrow w\alpha\beta \rightarrow_{\text{lm}}^* wa\gamma\} \cup \\ \cup \{\$ \mid S \rightarrow_{\text{lm}}^* wA \rightarrow w\alpha \rightarrow_{\text{lm}}^* w\}$$

nel secondo caso tipicamente quando $\alpha = \varepsilon$, cioè la produzione è $A \rightarrow \varepsilon$

Insiemi guida

Data una grammatica G , l'**insieme guida** di una produzione della grammatica $A \rightarrow \alpha$ - **Gui ($A \rightarrow \alpha$)** - è l'insieme dei terminali (o ε se ci si trova a fine parola) con cui *iniziano* le stringhe generabili a partire dalla produzione stessa :

$$\text{Gui}(A \rightarrow \alpha) = \{a \mid S \xrightarrow{\text{Im}}^* wA\beta \rightarrow w\alpha\beta \xrightarrow{\text{Im}}^* wa\gamma\} \cup \\ \cup \{\$ \mid S \xrightarrow{\text{Im}}^* wA \rightarrow w\alpha \xrightarrow{\text{Im}}^* w\}$$

L'insieme $\text{Gui}(A \rightarrow \alpha)$ si può esprimere usando $F()$ e $\text{FW}()$:

$$\text{Gui}(A \rightarrow \alpha) = \begin{cases} F(\alpha) & \text{se } \alpha \text{ non è annullabile} \\ (F(\alpha) - \{\varepsilon\}) \cup \text{FW}(A) & \text{se } \alpha \text{ è annullabile} \end{cases}$$

Esempio:

$$\text{Gui}(Z \rightarrow d) = \{d\}$$

$$\text{Gui}(Z \rightarrow XYZ) = \{a, c, d\}$$

$$\text{Gui}(Y \rightarrow c) = \{c\}$$

$$\text{Gui}(Y \rightarrow \varepsilon) = \{a, c, d\}$$

$$\text{Gui}(X \rightarrow Y) = \{a, c, d\}$$

$$\text{Gui}(X \rightarrow a) = \{a\}$$

Grammatiche LL(1)

Una **grammatica** è **LL(1)** se per ogni non terminale A e per ogni coppia di produzioni $A \rightarrow \alpha$ e $A \rightarrow \beta$, gli insiemi guida sono disgiunti:

$$\text{Gui}(A \rightarrow \alpha) \cap \text{Gui}(A \rightarrow \beta) = \Phi$$

Esempio:

La grammatica $\langle \{S\}, \{ (, [, <,),], > \}, P = \{ S \rightarrow (S) \mid [S] \mid <S> \mid \varepsilon \}, S \rangle$ è LL(1).

$$F(S) = \{ (, [, < \}$$

$$FW(S) = \{ \$,),], > \}$$

$$\text{Gui}(S \rightarrow (S)) = \{ (\}$$

$$\text{Gui}(S \rightarrow [S]) = \{ [\}$$

$$\text{Gui}(S \rightarrow <S>) = \{ < \}$$

$$\text{Gui}(S \rightarrow \varepsilon) = \{ \$,),], > \}$$

Insiemi GUIDA: esempio di calcolo

$$Z \rightarrow d \mid XYZ$$

$$Y \rightarrow c \mid \varepsilon$$

$$X \rightarrow Y \mid a$$

$$F(Z) = \{a, c, d\} \quad F(X) = \{a, c, \varepsilon\} \quad F(Y) = \{c, \varepsilon\}$$

$$Fw(Z) = \{\$, \}, \quad Fw(X) = \{a, c, d\}, \quad Fw(Y) = \{a, c, d\}$$

$$\text{Gui}(Z \rightarrow d) = F(d) = \{d\}$$

$$\begin{aligned} \text{Gui}(Z \rightarrow XYZ) &= F(XYZ) = (F(X) - \{\varepsilon\}) \cup F(YZ) = \\ &= (F(X) - \{\varepsilon\}) \cup (F(Y) - \{\varepsilon\}) \cup F(Z) = \{a, c, d\} \end{aligned}$$

$$\text{Gui}(Y \rightarrow c) = \{c\}$$

$$\text{Gui}(Y \rightarrow \varepsilon) = (F(\varepsilon) - \{\varepsilon\}) \cup Fw(Y) = \{a, c, d\}$$

$$\text{Gui}(X \rightarrow Y) = \{a, c, d\}$$

$$\text{Gui}(X \rightarrow a) = \{a\}$$

**Data la
seguente
Grammatica:**

Produzione

Insieme guida

- | | |
|--------------------------------|---------------|
| 1. $S \rightarrow PQ$ | $\{a, b, c\}$ |
| 2. $Q \rightarrow \&PQ$ | $\{\&\}$ |
| 3. $Q \rightarrow \varepsilon$ | $\{\$\}$ |
| 4. $P \rightarrow aPb$ | $\{a\}$ |
| 5. $P \rightarrow bPa$ | $\{b\}$ |
| 6. $P \rightarrow c$ | $\{c\}$ |

Verificare gli insiemi guida calcolando I FIRST e i FOLLOW usando direttamente le definizioni.