Linguaggi e Traduttori

Prof. Marco Gavanelli

29 giugno 2016

Esercizio 1 (4 punti)

Si consideri la seguente grammatica

$$\begin{array}{rlcrl} S & \to KX & \qquad & K & \to a \mid b \\ aX & \to aYb & & Y & \to aYb \mid a \\ bX & \to bYa & & & \end{array}$$

Si classifichi la grammatica secondo Chomsky.

Si scriva il linguaggio L_S generato dallo scopo S.

Si scriva una grammatica di tipo 2 o di tipo 3 non ambigua ed equivalente alla grammatica data.

Si disegni l'albero di derivazione della frase `baaba` nella nuova grammatica equivalente.

Esercizio 2 (5 punti)

Si consideri la seguente grammatica:

$$\begin{array}{rl} S & \to Ax \mid yB \\ B & \to \epsilon \mid zB \\ A & \to \epsilon \mid BaS \end{array}$$

1. Si classifichi la grammatica secondo Chomsky.
2. La grammatica è LL(1)? Se sì, si scriva la parsing table del PDA riconoscitore. Se no, si motivi il perché.

Qualora nei punti precedenti si sia riusciti a ottenere un riconoscitore, si mostri come l'automa riconosce le stringhe `ayzx` e `zaxx`, mostrando l'evoluzione dello stack.

Esercizio 3 (5 punti)

Si consideri la seguente grammatica:

$$\begin{array}{rl} S & \to Ba \mid Sa \mid b \\ B & \to Bb \mid Cb \mid Cc \\ C & \to Ca \mid b \mid Bc \mid Sc \end{array}$$

Si disegni l'automa riconoscitore del linguaggio generato. L'automa è deterministico? Se non lo è, si disegni un automa deterministico equivalente, evidenziando gli stati iniziali e finali.

Soluzione 1

La grammatica è di tipo 1 (dipendente dal contesto), come si vede dalle regole $aX \to aYb$ e $bX \to bYa$.

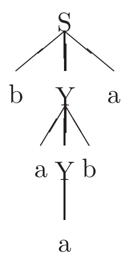
Il linguaggio generato è $L_S = \{a^{n+1}b^n \mid n \geq 1\} \cup \{ba^{m+1}b^m a \mid m \geq 0\}$.

Lo stesso linguaggio può essere generato dalla seguente grammatica di tipo 2 (indipendente dal contesto):

$$\begin{aligned} S &\to aYb \mid bYa \\ Y &\to aYb \mid a \end{aligned}$$

mentre è impossibile scrivere una grammatica regolare (di tipo 3) equivalente, visto che è necessario memorizzare il numero di simboli a, poiché il numero di simboli b dipende dal numero delle a.

L'albero di derivazione:



Soluzione 2

La grammatica è di tipo 2 (context-free).

Per verificare se la grammatica è di LL(1), scriviamo gli insiemi FIRST e FOLLOW dei simboli non terminali

	$FIRST$	$FOLLOW$
S	$\{y, z, a, x\}$	$\{\$, x\}$
B	$\{\epsilon, z\}$	$\{\$, a, x\}$
A	$\{\epsilon, z, a\}$	$\{x\}$

La parsing table è la seguente:

	a	x	y	z	$\$$
A	$A \to BaS$	$A \to \epsilon$		$A \to BaS$	
B	$B \to \epsilon$	$B \to \epsilon$		$B \to zB$	$B \to \epsilon$
S	$S \to Ax$	$S \to Ax$	$S \to yB$	$S \to Ax$	

La parsing table non presenta conflitti, per cui la grammatica è LL(1).

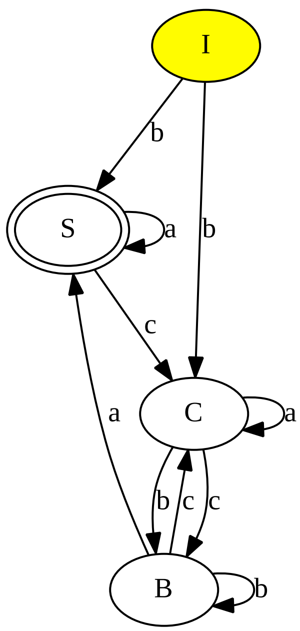
Riconoscimento stringhe:

Input	Stack
ayzx$	S$
ayzx$	Ax$
ayzx$	BaSx$
ayzx$	aSx$
yzx$	Sx$
yzx$	yBx$
zx$	Bx$
zx$	zBx$
x$	Bx$
x$	x$
$	$

Input	Stack
zaxx$	S$
zaxx$	Ax$
zaxx$	BaSx$
zaxx$	zBaSx$
axx$	BaSx$
axx$	aSx$
xx$	Sx$
xx$	Axx$
xx$	xx$
x$	x$
$	$

Soluzione 3

L'automa riconoscitore:



Questo automa non è deterministico: dallo stato I con input b si può passare in S o in C.

Per calcolare un automa deterministico, scriviamo la tabella di transizione, indicando con E lo stato di errore e aggiungendo stati che corrispondono a insiemi di stati dell'automa originario

	a	b	c
S	S	E	C
B	S	B	C
C	C	B	B
I	E	SC	E
SC	SC	BE	BC
BE	SE	BE	CE
BC	SC	B	BC
SE	SE	E	CE
CE	CE	BE	BE