Esercizi grammatiche context free

a.a. 2020-2021

Corso di Fondamenti di Informatica - 1 modulo

Corso di Laurea in Informatica

Università di Roma “Tor Vergata”

Prof. Giorgio Gambosi

(Esame 30-1-2020)
Si definisca una grammatica in CNF equivalente alla seguente

$$
\begin{aligned}
S &\longrightarrow ABa \\
A &\longrightarrow aAbb|\varepsilon \\
B &\longrightarrow bB|A|b
\end{aligned}
$$

1: soluzione

(Esame 30-1-2020)
Si definisca una grammatica in CNF equivalente alla seguente

$$
\begin{aligned}
S &\longrightarrow ABa \\
A &\longrightarrow aAbb \mid \varepsilon \\
B &\longrightarrow bB \mid A \mid b
\end{aligned}
$$

1: soluzione

A e B sono simboli annullabili, per cui l'eliminazione delle ε-produzioni fornisce

$$\begin{array}{rcl} S & \longrightarrow & ABa|Aa|Ba|a \\ A & \longrightarrow & aAbb|abb \\ B & \longrightarrow & bB|A|b \end{array}$$

1: soluzione

L'eliminazione della produzione unitaria $B \longrightarrow A$ fornisce

$$\begin{aligned} S &\longrightarrow ABa|Aa|Ba|a \\ A &\longrightarrow aAbb|abb \\ B &\longrightarrow bB|aAbb|abb|b \end{aligned}$$

1: soluzione

Tutti i simboli sono fecondi e raggiungibili, per cui non ci sono simboli inutili.

Una grammatica CNF equivalente è allora ottenuta dapprima eliminando i simboli terminali nelle parti destre delle produzioni non unitarie, ottenenendo

$$
\begin{array}{rcl}
S & \longrightarrow & ABX|AX|BX|a \\
A & \longrightarrow & XAYY|XYY \\
B & \longrightarrow & YB|AXAYY|XYY|b \\
X & \longrightarrow & a \\
Y & \longrightarrow & b
\end{array}
$$

1: soluzione

ed eliminando poi le produzioni con parti destre di lunghezza maggiore di 2, da cui

$$
\begin{array}{rcl}
S & \longrightarrow & UX|AX|BX|a \\
A & \longrightarrow & WZ|XZ \\
B & \longrightarrow & YB|VZ|XZ|b \\
Z & \longrightarrow & YY \\
W & \longrightarrow & XA \\
U & \longrightarrow & AB \\
V & \longrightarrow & AW \\
X & \longrightarrow & a \\
Y & \longrightarrow & b
\end{array}
$$

(Esame 13-2-2020)

Si definisca una grammatica context free che generi il linguaggio $L = \{a^r b^s c^t a^m c^n | s = r + t; n \geq 2m; r, t, m, n \geq 0\}$.

2: soluzione

(Esame 13-2-2020)
Si definisca una grammatica context free che generi il linguaggio
$L = \{a^r b^s c^t a^m c^n | s = r + t; n \geq 2m; r, t, m, n \geq 0\}$.

Una possibile soluzione è la grammatica

$$
\begin{aligned}
S &\longrightarrow ABCD \\
A &\longrightarrow aAb \mid \varepsilon \\
B &\longrightarrow bBc \mid \varepsilon \\
C &\longrightarrow aCcc \mid \varepsilon \\
D &\longrightarrow cD \mid \varepsilon
\end{aligned}
$$

(Esame 21-2-2018)

Si definisca una grammatica context free che generi il linguaggio
$L = \{a^r b^s c^t a^n c^n | s = r + t, r, t, n \geq 0\}$.

3: soluzione

(Esame 21-2-2018)
Si definisca una grammatica context free che generi il linguaggio
$L = \{a^r b^s c^t a^n c^n | s = r + t, r, t, n \geq 0\}$.

Una possibile soluzione è la grammatica

$$
\begin{array}{rcl}
S & \longrightarrow & ABC \\
A & \longrightarrow & aAb \mid \varepsilon \\
B & \longrightarrow & bBc \mid \varepsilon \\
C & \longrightarrow & aCc \mid \varepsilon
\end{array}
$$

(Esame 9-2-2018)
Ridurre la grammatica seguente in GNF

$$
\begin{array}{rcl}
S & \longrightarrow & aEb \mid aaC \mid AA \\
A & \longrightarrow & BC \mid bS \mid b \\
B & \longrightarrow & aB \mid \varepsilon \\
C & \longrightarrow & Ca \mid Cb \\
D & \longrightarrow & a \mid c
\end{array}
$$

4: soluzione

(Esame 9-2-2018)
Ridurre la grammatica seguente in GNF

$$
\begin{array}{rcl}
S & \longrightarrow & aEb \mid aaC \mid AA \\
A & \longrightarrow & BC \mid bS \mid b \\
B & \longrightarrow & aB \mid \varepsilon \\
C & \longrightarrow & Ca \mid Cb \\
D & \longrightarrow & a \mid c
\end{array}
$$

4: soluzione

Per portare la grammatica in forma ridotta eliminiamo l'ε-produzione, ottenendo

$$\begin{array}{rcl} S & \longrightarrow & aEb \mid aaC \mid AA \\ A & \longrightarrow & BC \mid C \mid bS \mid b \\ B & \longrightarrow & aB \mid a \\ C & \longrightarrow & Ca \mid Cb \\ D & \longrightarrow & a \mid c \end{array}$$

4: soluzione

Eliminiamo quindi la produzione unitaria

$$
\begin{aligned}
S &\longrightarrow aEb \mid aaC \mid AA \\
A &\longrightarrow BC \mid Ca \mid Cb \mid bS \mid b \\
B &\longrightarrow aB \mid a \\
C &\longrightarrow Ca \mid Cb \\
D &\longrightarrow a \mid c
\end{aligned}
$$

4: soluzione

Osserviamo ora che C e E sono simboli non fecondi, per cui eliminandoli otteniamo

$$\begin{aligned}
S &\longrightarrow AA \\
A &\longrightarrow bS \mid b \\
B &\longrightarrow aB \mid a \\
D &\longrightarrow a \mid c
\end{aligned}$$

4: soluzione

a questo punto, eliminando i simboli non raggiungibili B e D, otteniamo la grammatica equivalente in forma ridotta

$$\begin{aligned} S &\longrightarrow AA \\ A &\longrightarrow bS \mid b \end{aligned}$$

4: soluzione

La corrispondente grammatica in CNF è

$$\begin{aligned} S &\longrightarrow AA \\ A &\longrightarrow BS \mid b \\ B &\longrightarrow b \end{aligned}$$

4: soluzione

e da questa la grammatica in GNF

$$\begin{aligned} S &\longrightarrow bSA \mid bA \\ A &\longrightarrow bS \mid b \\ B &\longrightarrow b \end{aligned}$$

(Esame 24-9-2019)
Una espressione parentetica corretta è una stringa non nulla sull'alfabeto $\Sigma = \{(,)\}$ tale che

1. il numero di caratteri "(" è uguale al numero di caratteri ")"
2. per ogni prefisso x di σ, il numero di caratteri "(" è maggiore o uguale al numero di caratteri ")"

Definire una grammatica in GNF che generi il linguaggio delle espressioni parentetiche corrette.

5: soluzione

(Esame 24-9-2019)
Una espressione parentetica corretta è una stringa non nulla sull'alfabeto $\Sigma = \{(,)\}$ tale che

1. il numero di caratteri "(" è uguale al numero di caratteri ")"
2. per ogni prefisso x di σ, il numero di caratteri "(" è maggiore o uguale al numero di caratteri ")"

Definire una grammatica in GNF che generi il linguaggio delle espressioni parentetiche corrette.

5: soluzione

$$S \quad \longrightarrow \quad (S) \mid SS \mid ()$$

che risulta già in forma ridotta

5: soluzione

In CNF:

$$
\begin{aligned}
S &\rightarrow SS \mid XQ \mid PQ \\
X &\rightarrow PS \\
P &\rightarrow (\\
Q &\rightarrow)
\end{aligned}
$$

5: soluzione

In GNF:

Step 1 (eliminazione ricorsione sx, $\alpha_1 = S, \beta_1 = XQ, \beta_2 = PQ$)

$$
\begin{aligned}
S &\rightarrow XQ \mid PQ \mid XQZ \mid PQZ \\
X &\rightarrow PS \\
P &\rightarrow (\\
Q &\rightarrow) \\
Z &\rightarrow S \mid SZ
\end{aligned}
$$

5: soluzione

Step 2 (sostituzioni produzioni su terminali originari):

$$
\begin{array}{rcl}
S & \longrightarrow & (SQ \mid (Q \mid (SQZ \mid (QZ \\
X & \longrightarrow & (S \\
P & \longrightarrow & (\\
Q & \longrightarrow &) \\
Z & \longrightarrow & S \mid SZ
\end{array}
$$

5: soluzione

Step 3 (sostituzioni produzioni su nuovi terminali):

$$
\begin{array}{rcl}
S & \longrightarrow & (SQ \mid (Q \mid (SQZ \mid (QZ \\
X & \longrightarrow & (S \\
P & \longrightarrow & (\\
Q & \longrightarrow &) \\
Z & \longrightarrow & (SQ \mid (Q \mid (SQZ \mid (QZ \mid (SQZ \mid (QZ \mid (SQZZ \mid (QZZ
\end{array}
$$

5: soluzione

ed eliminando i simboli che risultano inutili,

$$
\begin{aligned}
S &\rightarrow (SQ \mid (Q \mid (SQZ \mid (QZ \\
Q &\rightarrow\ ) \\
Z &\rightarrow (SQ \mid (Q \mid (SQZ \mid (QZ \mid (SQZ \mid (QZ \mid (SQZZ \mid (QZZ
\end{aligned}
$$

(Esame 13-9-2019)
Definire una grammatica context free che generi l'insieme delle stringhe su $\{a, b, c\}$ descritte dall'espressione (non regolare) $(((a^n b^n) + (b^m a^m))c)^*$

6: soluzione

(Esame 13-9-2019)
Definire una grammatica context free che generi l'insieme delle stringhe su $\{a, b, c\}$ descritte dall'espressione (non regolare) $(((a^n b^n) + (b^m a^m))c)^*$

$$
\begin{array}{rcl}
S & \longrightarrow & SC|\varepsilon \\
A & \longrightarrow & aAb|\varepsilon \\
B & \longrightarrow & bBa|\varepsilon \\
C & \longrightarrow & Ac|Bc
\end{array}
$$

(Esame 13-9-2019)
Definire formalmente il linguaggio comprendente tutte e sole le espressioni regolari sull'alfabeto $\Sigma = \{a, b, c\}$

7: soluzione

(Esame 13-9-2019)
Definire formalmente il linguaggio comprendente tutte e sole le espressioni regolari sull'alfabeto $\Sigma = \{a, b, c\}$

$$S \quad \longrightarrow \quad (S+S)|(SS)|(S)^*|a|b|c|\varepsilon$$

Si definisca una grammatica context free che generi il linguaggio $L = \{a^r b^s c^t | t = r - s\}$.

8: soluzione

Si definisca una grammatica context free che generi il linguaggio $L = \{a^r b^s c^t | t = r - s\}$.

8: soluzione

Osservando che $r = s + t$, una possibile grammatica che generi L è ad esempio:

$$
\begin{array}{rcl}
S & \rightarrow & aSc|U \\
U & \rightarrow & aUb|\varepsilon
\end{array}
$$

Sia L il linguaggio generato dalla seguente grammatica context free

$$S \quad \rightarrow \quad \varepsilon | 0S1S | 1S0S$$

derivare una grammatica in Forma Normale di Greibach che generi $L - \{\varepsilon\}$.

9: soluzione

Sia L il linguaggio generato dalla seguente grammatica context free

$$S \rightarrow \varepsilon | 0S1S | 1S0S$$

derivare una grammatica in Forma Normale di Greibach che generi $L - \{\varepsilon\}$.

9: soluzione

Il primo passo prevede la derivazione della grammatica in forma ridotta equivalente.

Eliminazione delle ε produzioni:

$$S \quad \rightarrow \quad 0S1S|1S0S|01S|0S1|01|1S0|10S|10$$

Non ci sono produzioni unitarie o simboli inutili.

9: soluzione

Forma normale di Chomsky:

$$
\begin{aligned}
S &\rightarrow XY|YX|ZY|XU|ZU|YZ|UX|UZ \\
X &\rightarrow ZS \\
Y &\rightarrow US \\
Z &\rightarrow 0 \\
U &\rightarrow 1
\end{aligned}
$$

9: soluzione

Forma normale di Greibach:

- dopo la prima fase

$$
\begin{aligned}
S &\rightarrow XY|YX|ZY|XU|ZU|YZ|UX|UZ \\
X &\rightarrow ZS \\
Y &\rightarrow US \\
Z &\rightarrow 0 \\
U &\rightarrow 1
\end{aligned}
$$

- dopo la seconda fase

$$
\begin{aligned}
S &\rightarrow 0SY|1SX|0Y|1U|0U|0Z|1X|1Z \\
X &\rightarrow 0S \\
Y &\rightarrow 1S \\
Z &\rightarrow 0 \\
U &\rightarrow 1
\end{aligned}
$$

Si consideri il linguaggio

$$L = \{w \in \{a, b, c\}^+, \#_a(w) = \#_b(w) = \#_c(w)\}$$

dove con $\#_c(x)$ indichiamo il numero di occorrenze del carattere c nella stringa x.

Si mostri che L non è context free.

10: soluzione

Si consideri il linguaggio

$$L = \{w \in \{a, b, c\}^+, \#_a(w) = \#_b(w) = \#_c(w)\}$$

dove con $\#_c(x)$ indichiamo il numero di occorrenze del carattere c nella stringa x.

Si mostri che L non è context free.

10: soluzione

Possiamo applicare il pumping lemma, considerando ad esempio, dato $n > 0$, la stringa $\sigma = a^n b^n c^n$.

Per qualunque decomposizione $\sigma = uvwxy$ con $|vwx| \leq n$ si deve necessariamente avere che vwx (e quindi vx) non può contenere sia caratteri a che caratteri b che caratteri c. Inoltre, per costruzione, $|vx| \geq 1$.

Consideriamo ad esempio il caso in cui $\#_a(vx) = 0$: allora avremo, relativamente a $\sigma' = uv^2wx^2y$, che $\#_a(\sigma') = \#_a(\sigma)$, $\#_b(\sigma') = \#_b(\sigma) + \#_b(vx)$ e $\#_c(\sigma') = \#_c(\sigma) + \#_c(vx)$, in cui $\#_b(vx) + \#_c(vx) > 0$. Ne deriva che $\sigma' \notin L$, e quindi che L non è context free. Lo stesso chiaramente vale se assumiamo $\#_b(vx) = 0$ o $\#_c(vx) = 0$.

Sia dato il linguaggio $L \subseteq \{a, b, c\}^*$ tale che $w \in L$ se e solo se $\#_w(a) = \#_w(c)$, dove $\#_w(x)$ indica il numero di occorrenze di $x \in \{a, b, c\}$ in w. Tale linguaggio è context free? Motivare la propria risposta o mediante applicazione del pumping lemma o fornendo una grammatica CF che lo generi .

11: soluzione

Sia dato il linguaggio $L \subseteq \{a, b, c\}^*$ tale che $w \in L$ se e solo se $\#_w(a) = \#_w(c)$, dove $\#_w(x)$ indica il numero di occorrenze di $x \in \{a, b, c\}$ in w. Tale linguaggio è context free? Motivare la propria risposta o mediante applicazione del pumping lemma o fornendo una grammatica CF che lo generi .

11: soluzione

Il linguaggio è context-free: per motivare tale risposta definiamo una grammatica CF che lo generi

$$S \quad \longrightarrow \quad \varepsilon | bS | aScS | cSaS$$

Si definisca una grammatica context free che generi il linguaggio

$$L = \{a^m b^n + a^r b^s a^t | 1 \leq m \leq n \leq 3m; s \geq 1, 1 \leq r \leq t \leq 2r\}$$

12: soluzione

Si definisca una grammatica context free che generi il linguaggio

$$L = \{a^m b^n + a^r b^s a^t | 1 \leq m \leq n \leq 3m; s \geq 1, 1 \leq r \leq t \leq 2r\}$$

12: soluzione

$$
\begin{array}{rcl}
S & \longrightarrow & S_1|S_2 \\
S_1 & \longrightarrow & ab|abb|abbb|aS_1b|aS_1bb|aS_1bbb \\
S_2 & \longrightarrow & aBa|aBaa|aS_2a|aS_2aa \\
B & \longrightarrow & bB|b
\end{array}
$$

Verificare se il linguaggio

$$L = \{a^i b^j c^k | i < j \wedge i < k\}$$

è context free o meno.

13: soluzione

Verificare se il linguaggio

$$L = \{a^i b^j c^k | i < j \wedge i < k\}$$

è context free o meno.

13: soluzione

Applicando il pumping lemma per i CFL, abbiamo che se L è CF esiste un n tale che per $i+j+k>n$ possiamo scrivere $z = uvwxy$ con $|vx| > 1$ e $|vwx| <\leq n$, e che $uv^i wx^i y \in L$ per ogni $i \geq 0$.

Consideriamo la stringa $a^m b^{m+1} c^{m+1}$, con $n = 3m+2$, e osserviamo che per qualunque decomposizione $z = uvwxy$:

- se v o x corrispondono a sequenze non omogenee (a^r, b^s, c^t), allora $uv^2wx^2y \notin L$
- altrimenti, se $v = a^r$ e $x = b^s$, se $r > 0$ la stringa $uv^2wx^2y \notin L$ in quanto il numero di a è maggiore del numero di c; se $r = 0$ la stringa $uwy \notin L$ in quanto il numero di b è minore o uguale del numero di a. Le stesse considerazioni valgono se $v = a^r$ e $x = c^s$.
- infine, se $v = b^r$ e $x = c^s$, la stringa $uwy \notin L$ in quanto il numero di a è maggiore o uguale di almeno uno tra il numero di b e il numero di c;

Sia dato il linguaggio $L \subseteq \{a, b, c\}^*$ tale che $w \in L$ se e solo se $\#_w(a) = \#_w(c)$, dove $\#_w(x)$ indica il numero di occorrenze di $x \in \{a, b, c\}$ in w.

Tale linguaggio è context free?

Motivare la propria risposta o mediante applicazione del pumping lemma o fornendo una grammatica CF che lo generi .

14: soluzione

Sia dato il linguaggio $L \subseteq \{a, b, c\}^*$ tale che $w \in L$ se e solo se $\#_w(a) = \#_w(c)$, dove $\#_w(x)$ indica il numero di occorrenze di $x \in \{a, b, c\}$ in w.

Tale linguaggio è context free?

Motivare la propria risposta o mediante applicazione del pumping lemma o fornendo una grammatica CF che lo generi .

14: soluzione

Il linguaggio è context-free: per motivare tale risposta definiamo una grammatica CF che lo generi

$$S \quad \longrightarrow \quad \varepsilon|bS|Sb|aScS|cSaS$$

Mostrare se il seguente linguaggio è o meno context free:

$L = \{w_1w_2w_3 : w_1 \in \{a, b\}^+, w_2 \in \{c, d\}^+, w_3 \in \{e, f\}^+, |w_1| = |w_2| = |w_3|\}$

15: soluzione

Mostrare se il seguente linguaggio è o meno context free:

$$L = \{w_1 w_2 w_3 : w_1 \in \{a, b\}^+, w_2 \in \{c, d\}^+, w_3 \in \{e, f\}^+, |w_1| = |w_2| = |w_3|\}$$

15: soluzione

Il linguaggio non è context-free. Per dimostrare ciò utilizziamo il pumping lemma per i linguaggi di tipo 2.

Dato n, consideriamo la stringa $\sigma = a^n c^n e^n$. Se consideriamo le decomposizioni $\sigma = uvwxy$ con $|vwx| \leq n$ e $|vx| \geq 1$ si hanno due casi possibili:

- sia v che x sono sequenze di stessi caratteri (ad esempio $v = a^k$ e x=c^h): si osservi che in tal caso uno dei tre caratteri che compaiono in σ non compare in vx. Di conseguenza la stringa $\sigma' = uv^2wx^2y$ non presenta lo stesso numero di a, c ed e, e quindi non appartiene al linguaggio. Si osservi che come caso particolare si ha $v = \varepsilon$ o $x = \varepsilon$: la conclusione deriva anche in questo caso.
- almeno una tra v e x non è una sequenza di stessi caratteri (ad esempio, $v = a^h c^k$): in tal caso, $v^2 = a^h c^k a^h c^k$ e $\sigma' = uv^2wx^2y$ non appartiene al linguaggio.

Si consideri il linguaggio

$$L = \{0^i 1^j | i \geq j\}$$

Il linguaggio è regolare o context free? Dimostrare quale delle due affermazioni è vera. Si definisca inoltre una grammatica (di tipo 3 o di tipo 2, rispettivamente) che generi tutte e sole le stringhe del linguaggio.

16: soluzione

Si consideri il linguaggio

$$L = \{0^i 1^j | i \geq j\}$$

Il linguaggio è regolare o context free? Dimostrare quale delle due affermazioni è vera. Si definisca inoltre una grammatica (di tipo 3 o di tipo 2, rispettivamente) che generi tutte e sole le stringhe del linguaggio.

16: soluzione

Il linguaggio non è regolare, ma è context free. Per verificare che non è regolare si può utilizzare il pumping lemma applicato (fissato n)alla stringa $0^n 1^n \in L$. Dato che per ogni $uvx = 0^n 1^n$ con $|uv| \leq n$ e $|v| \geq 1$ si deve avere necessariamente che $v = 0^k$ per un qualche $k > 0$, si che $uv^0 w = uv = 0^{n-k} 1^k \notin L$, per cui L non è regolare.

Una grammatica CF che genera L è ad esempio

$$\begin{aligned} S &\longrightarrow 0S1|0T1|\varepsilon \\ T &\longrightarrow 0T|0 \end{aligned}$$

Utilizzare il *pumping lemma* per dimostrare che il linguaggio $L = \{a^n b a^{2n} b a^{3n} \mid n \geq 0\}$ su $\Sigma = \{a, b\}$ non è context free.

Utilizzare il *pumping lemma* per dimostrare che il linguaggio
$L = \{a^n b^n c^i \mid i \leq n\}$ su $\Sigma = \{a, b, c\}$ non è context free.

Utilizzare il *pumping lemma* per dimostrare che il linguaggio $L = \{a^i b^j c^k \mid 0 \leq i < j < k\}$ su $\Sigma = \{a, b, c\}$ non è context free.

Utilizzare il *pumping lemma* per dimostrare che il linguaggio
$L = \{a^n b^n c^m \mid n \leq m \leq 2n$ su $\Sigma = \{a, b, c\}$ non è context free.

Definire un automa a pila che accetti il linguaggio

$$L = \{a^n b^m | 1 \leq n \leq m\}$$

per pila vuota.

21: soluzione

- L'automa legge la sequenza iniziale di a ponendo sulla pila un simbolo A per ogni simbolo letto.
- L'automa cambia stato per leggere la sequenza di b, eliminando i caratteri A dalla pila.
- Se si raggiunge il fondo della pila (il simbolo Z_0) la stringa va accettata, completando la lettura degli eventuali b mancanti ed eliminando poi Z_0.

21: soluzione

	(q_0, Z_0)	(q_0, A)	(q_1, Z_0)	(q_1, A)
a	(q_0, AZ_0)	(q_0, AA)	-	-
b	-	(q_1, ε)	(q_1, Z_0)	(q_1, ε)
ε	-	-	(q_1, ε)	-

Definire un automa a pila che accetti il linguaggio

$$L = \{w \in \{a, b\}^+ | \#_a(w) \geq \#_b(w)\}$$

Dove $\#_c(x)$ indica il numero di occorrenze del carattere c nella stringa x.

22: soluzione

- L'automa manitine traccia, sulla pila, della differenza tra il numero di caratteri a e il numero di caratteri b letti fino a ora (o vice versa, a seconda che siano stati letti più a o più b).
- La stringa è accettata se al termine della sua lettura la pila è vuota o contiene tutti simboli A.
- Per accettare per pila vuota l'automa prevede che in qualunque istante in cui il numero di a lette è almeno pari al numero di b possa entrare in uno stato q_1 di svuotamento della pila.

22: soluzione

	(q_0, Z_0)	(q_0, A)	(q_0, B)	(q_1, Z_0)	(q_1, A)
a	(q_0, AZ_0)	(q_0, AA)	(q_0, ε)	-	-
b	(q_0, BZ_0)	(q_0, ε)	(q_0, BB)	-	-
ε	(q_1, ε)	(q_1, ε)	-	(q_1, ε)	(q_1, ε)

Si definisca un automa a pila (eventualmente non deterministico) che accetti il linguaggio $L = \{a^r b^s c^t a^n c^n | s = r + t, r, t, n \geq 0\}$.

Si consideri il linguaggio $L \subset \{0, 1\}^*$ tale che $\sigma \in L$ se e solo se $\#_0(\sigma) = \#_1(\sigma)$, dove $\#_a(s)$ indica il numero di occorrenze del carattere a nella stringa s. Si definisca una grammatica CF in GNF che generi L.

24: soluzione

Una possibile grammatica è la seguente:

$$S \quad \longrightarrow \quad 0S1S|1S0S|\varepsilon$$

L'eliminazione della ε-produzione porta alla grammatica equivalente

$$S \quad \longrightarrow \quad 0S1S|1S0S|0S1|01S|1S0|10S|01|10$$

che non presenta produzioni unitarie o simboli inutili.

24: soluzione

La grammatica in CNF che ne deriva è

$$\begin{array}{rcl} S & \rightarrow & XY|YX|XU|ZY|YZ|UX|ZU|UZ \\ X & \rightarrow & ZS \\ Y & \rightarrow & US \\ Z & \rightarrow & 0 \\ U & \rightarrow & 1 \end{array}$$

24: soluzione

da cui deriva immediatamente la grammatica in GNF

$$\begin{aligned}
S &\rightarrow 0SY|1SX|0SU|0Y|1SZ|1X|0U|1Z \\
X &\rightarrow 0S \\
Y &\rightarrow 1S \\
Z &\rightarrow 0 \\
U &\rightarrow 1
\end{aligned}$$

Si consideri il linguaggio

$$L = \{0^i 1^j | i \geq j\}$$

Il linguaggio è regolare o context free? Dimostrare quale delle due affermazioni è vera. Si definisca inoltre una grammatica (di tipo 3 o di tipo 2, rispettivamente) che generi tutte e sole le stringhe del linguaggio.

25: soluzione

Il linguaggio non è regolare, ma è context free. Per verificare che non è regolare si può utilizzare il pumping lemma applicato (fissato n)alla stringa $0^n 1^n \in L$. Dato che per ogni $uvx = 0^n 1^n$ con $|uv| \leq n$ e $|v| \geq 1$ si deve avere necessariamente che $v = 0^k$ per un qualche $k > 0$, si che $uv^0 w = uv = 0^{n-k} 1^k \notin L$, per cui L non è regolare.

25: soluzione

Una grammatica CF che genera L è ad esempio

$$\begin{aligned} S &\longrightarrow 0S1|0T1|\varepsilon \\ T &\longrightarrow 0T|0 \end{aligned}$$

Definire un automa a pila che accetta per stato finale il linguaggio composto dalle stringhe $w \in \{0, 1\}^+$ contenenti uno stesso numero di 0 e di 1.

26: soluzione

Un possibile automa ha 2 soli stati q_0, q_F e un alfabeto di pila Z_0, Z, U. Ad ogni istante la pila contiene, al di sopra di Z_0, una sequenza di Z di dimensione pari a $\#(0) - \#(1)$ se $\#(0) - \#(1) > 0$ o una sequenza di U di dimensione pari a $\#(1) - \#(0)$ se $\#(0) - \#(1) < 0$.

	$(q_0, 0)$	$(q_0, 1)$	(q_0, ε)
Z_0	(q_0, ZZ_0)	(q_0, UZ_0)	(q_F, ε)
Z	(q_0, ZZ)	(q_0, ε)	-
U	(q_0, ε)	(q_0, UU)	-

Definire un automa a pila che accetti il seguente linguaggio

$$L = \{a^p b^{p+2q} a^q ; p, q > 0\}$$

descrivendo e giustificando le scelte effettuate.

27: soluzione

- Nello stato q_0 vengono posti nella pila tanti simboli A quanti simboli a sono letti. Lo stato diventa q_1 al primo simbolo b letto
- Nello stato q_1, un simbolo A viene tolto dalla pila per ogni b letto, fino a giungere al fondo della pila e passare in q_2.
- In q_2, per ogni simbolo b letto viene posto sulla pila un simbolo B. L'automa passa in q_3 quando legge un nuovo simbolo a
- In q_3, per ogni simbolo a letto l'automa dovrà togliere due simboli B: per far ciò, passerà ciclicamente in q_3, in cui toglierà la prima B dalla pila avendo letto a, e in q_4, in cui toglierà la seconda B con una ε-transizione.
- Infine, se l'automa si trova in q_4, ed ha quindi tolto BB dalla pila avendo letto a, può eliminare Z_0 dalla pila con una ε. La stringa è accettata per pila vuota.

27: soluzione

	(q_0, Z_0)	(q_0, A)	(q_1, Z_0)	(q_1, A)	(q_2, B)	(q_3, B)	(q_4, Z_0)	(q_4, B)
a	(q_0, AZ_0)	(q_0, AA)	-	-	(q_3, ε)	-	-	(q_3, ε)
b	-	(q_1, ε)	(q_2, BZ_0)	(q_1, ε)	(q_2, BB)	-	-	-
ε	-	-	-	-	-	(q_4, ε)	(q_4, ε)	-

Si consideri il linguaggio

$$L = \{w\#x | w, x \in \{0, 1\}^+, w^R \text{ è suffisso di } x\}$$

Si verifichi che L è context free definendo un automa a pila che lo accetta.

28: soluzione

L'automa dapprima (nello stato q_0) legge w e la trascrive sulla pila in ordine inverso. Alla lettura del carattere # l'automa passa nello stato q_1 di lettura di x: in qualunque passo in cui il carattere letto corrisponde a quello in cima alla pila l'automa effettua una scelta non deterministica tra due opzioni:

1. assumere che w^R compaia in x a partire da questo carattere, in tal caso passa nello stato q_2 ed elimina il primo carattere dalla pila
2. assumere che w^R non compaia in x a partire da questo carattere, e continuare a leggere caratteri, nello stato q_1

Nello stato q_2, l'automa procede nella computazione fin tanto che i caratteri letti corrispondono a quelli via via estratti dalla pila. Nel caso positivo, la stringa termina con Z_0 sulla pila: questo carattere viene quindi estratto con una ε-transizione.

28: soluzione

	(q_0, Z_0)	(q_0, Z)	(q_0, U)	(q_1, Z)	(q_1, U)	(q_2, Z)	(q_2, U)	(q_2, Z_0)
0	(q_0, ZZ_0)	(q_0, ZZ)	(q_0, UZ)	$\{(q_1, Z), (q_2, \varepsilon)\}$	(q_1, U)	(q_2, ε)	-	-
1	(q_0, UZ_0)	(q_0, UZ)	(q_0, UU)	(q_1, Z)	$\{(q_1, U), (q_2, \varepsilon)\}$	-	(q_2, ε)	-
#	-	(q_1, Z)	(q_1, U)	-	-	-	-	-
ε	-	-	-	-	-	-	-	(q_2, ε)

Sia dato il linguaggio

$$L = \{a^n b^m c^k | k = |n - m|\}$$

Definire una grammatica context free che generi il linguaggio. Discutere se la grammatica risultante è ambigua.

29: soluzione

Una possibile grammatica è la seguente:

$$
\begin{array}{rcl}
S & \longrightarrow & S_1|S_3 \\
S_1 & \longrightarrow & aS_1b|S_2 \\
S_2 & \longrightarrow & aS_2c|\varepsilon \\
S_3 & \longrightarrow & S_4S_5 \\
S_4 & \longrightarrow & aS_4b|\varepsilon \\
S_5 & \longrightarrow & bS_5c|\varepsilon
\end{array}
$$

S_1 corrisponde al caso $n \geq m$, mentre S_3 al caso $m \geq n$.

29: soluzione

La grammatica in questo caso risulta ambigua, in quanto ad esempio la stringa *aabb* può essere generata sia come

$S \Rightarrow S_1 \Rightarrow aS_1b \Rightarrow aaS_1bb \Rightarrow aabb$ che come

$S \Rightarrow S_3 \Rightarrow S_4S_5 \Rightarrow aS_4bS_5 \Rightarrow aaS_4bbS_5 \Rightarrow aabbS_5 \Rightarrow aabb$

Dimostrare che il seguente linguaggio.

$$L = \{w \in \{a, b\}^+ : \#_w(a) = 2\#_w(b)\}$$

è context free, dove $\#_w(x)$ indica il numero di occorrenze del carattere x nella stringa w

30: soluzione

Una possibile soluzione è quella di definire un PDA che accetta il linguaggio.

	(q_0, Z_0)	(q_0, X)	(q_0, Y)	(q_1, Z_0)	(q_1, Y)
a	(q_0, XXZ_0)	(q_0, XXX)	(q_1, ε)	-	-
b	(q_0, YZ_0)	(q_0, ε)	(q_0, YY)	-	-
ε	(q_0, ε)	-	-	(q_0, X)	(q_0, ε)

Si definisca un automa a pila (eventualmente non deterministico) che accetti il linguaggio $L = \{a^r b^s c^t | t = r - s\}$.