Grammatiche generative
Esempi ed esercizi

Grammatica

Una grammatica è una quadrupla (V,T, S,P) dove:

1. V è un insieme finito non vuoto di simboli non terminali;
2. T è un insieme finito non vuoto di simboli terminali (NOTA V e T devono essere insiemi disgiunti);
3. S è un simbolo iniziale (anche detto simbolo di partenza oppure simbolo di frase) appartenente a V;
4. P è un insieme finito di produzioni della forma $\alpha \rightarrow \beta$ dove α e β sono sequenze di simboli terminali e non terminali α contiene almeno un simbolo non terminale.

Grammatica

Intuitivamente, una produzione $\alpha \rightarrow \beta$ significa che ogni occorrenza della sequenza alla sinistra della produzione (ovvero α) può essere sostituita con la sequenza alla destra (ovvero β).

In particolare, diremo

- che la stringa $\gamma\ \beta\ \delta$ è direttamente generabile dalla stringa $\gamma\ \alpha\ \delta$ se $\alpha \rightarrow \beta$ è una produzione della grammatica.
- le frasi del linguaggio associato a una grammatica sono generate partendo da S e applicando le produzioni fino a quando non restano solo simboli terminali.

Grammatica

Si tratta di una definizione ricorsiva: esiste un caso iniziale (la sequenza composta dal solo simbolo iniziale S) e poi si applicano ripetutamente le possibili sostituzioni indicate dalle produzioni. In particolare,

- diremo che la stringa di simboli terminali e non terminali γ è generabile da S, se esiste una sequenza di stringhe

  $S=\gamma_1, \gamma_2, \ldots \gamma_{(n-1)}, \gamma_n = \alpha,$

  tale che, per ogni i, $1 \leq i<n$, $\gamma_{(i+1)}$ è direttamente generabile da γ_i.
- L'insieme di tutte le sequenze di terminali T* che sono generabili dal simbolo iniziale S formano il linguaggio L(G) generato dalla grammatica G.

Esempi

Data la grammatica G=<{S,B,C} ,{a,b},S, P> con produzioni (lettere maiuscole Non term., minuscole Terminali)

S → aS   S→ B   (la scriviamo S → aS |B)

B → bB   S→ bC

C → cC   C→ c

la stringa aaabbC si ottiene per derivazione diretta da aaabB applicando la quarta produzione, cioè aaabB ⇒ aaabbB

la stringa aabbC si ottiene per derivazione da aS, cioè aS ⇒∗ aabbC

Esempi

La grammatica G=<{a,b},{S, A},P,S> con le produzioni:

$S \rightarrow aB \mid b$

$B \rightarrow aS$

genera il linguaggio $L = \{a^n b \mid n \geq 0 \text{ e pari}\}$.

Per n=0 abbiamo $S \rightarrow b$; quindi b appartien a L

Se invece usiamo $S \rightarrow aB$ poi deriviamo $B \rightarrow aS$ e otteniamo una nuova a all'inizio e avremo aaS;

ripetendo otteniamo il linguaggio $L = \{a^n b \mid n \geq 0 \text{ e pari}\}$

Esempi

La grammatica G=<{a,b,c},{S,A},P,S> con produzioni $S \rightarrow Ab$ $A \rightarrow Sa$
non genera alcuna stringa (cioè genera Λ, il linguaggio vuoto)

Ricorda le stringhe che appartengono al linguaggio sono le stringhe di soli simboli terminali

Nella grammatica data la parte destra di ciascuna produzione ha sempre un simbolo non terminale

Gerarchia di Chomsky

Grammatiche regolari o di tipo 3 In questo caso, ciascuna produzione della grammatica sono **lineari a destra** ovvero deve essere del tipo $A \to aB$ oppure del tipo $A \to a$[1].
Un linguaggio L **è regolare o di tipo 3** se esiste una grammatica di tipo 3 che lo genera.

Grammatiche libere da contesto o di tipo 2 In questo caso, ogni produzione della grammatica deve essere del tipo $A \to \alpha$.
Un linguaggio L **è libero dal contesto o di tipo 2** se esiste una grammatica di tipo 2 che lo genera.

Grammatiche contestuali o di tipo 1 In tal caso, ciascuna produzione della grammatica deve essere del tipo $\alpha \to \beta$ con $|\beta| \geqslant |\alpha|$.
Un linguaggio L **è contestuale o di tipo 1** se esiste una grammatica di tipo 1 che lo genera.

Grammatiche non limitate o di tipo 0 In tal caso, nessun vincolo sussiste sulla tipologia delle produzioni della grammatica.

Grammatica tipo 3 = linguaggi regolari

ESEMPIO: grammatica per la data
V (simb. nonterminali)= {S, Giorno,Mese, Anno, Cifra}
T (simb. Terminali) = {0,1,2,3,4,5,6,7,8,9,/}
Produzioni:

L'esempio precedente per definire una data porta alla seguente definizione di grammatica. $G = (V, T, S, P)$, dove $V = \{S, Giorno, Mese, Anno, Cifra\}$, $T = \{0,1,2,3,4,5,6,7,8,9,/\}$ e P contiene le seguenti produzioni:

$$S \rightarrow Giorno/Mese/Anno$$

$$Giorno \rightarrow Cifra/, Cifra$$

$$Mese \rightarrow Cifra/, Cifra$$

$$Anno \rightarrow Cifra/, Cifra/, Cifra/, Cifra$$

$$Cifra \rightarrow 0|1|2|3|4|5|6|7|8|9$$

Grammatiche tipo 2: linguaggi liberi dal contesto

ESEMPIO

Grammatica per $L = \{0^n1^n, n>0\}$

$V=\{S\}$, $T=\{0,1\}$

Produzioni $S \rightarrow 01$ $\quad$ $S \rightarrow 0S1$

Per brevità possiamo anche scrivere

$S \rightarrow 01 \mid 0S1$

Per ottenere 00001111 abbiamo

$S \rightarrow 0S1 \rightarrow 00S11 \rightarrow 000S111 \rightarrow 00001111$

Grammatiche tipo 1: linguaggi liberi dal contesto

ESEMPIO

Grammatica per $L = \{0^n 1^n 2^n, n>0\}$

$V=\{S,A,B\}$, $T=\{0,1,2\}$

Produzioni

$S \rightarrow 012 \quad S \rightarrow 0A12 \quad A1 \rightarrow 1A \quad A2 \rightarrow B122$

$1B \rightarrow B1 \quad 0B \rightarrow 00 \quad 0B \rightarrow 00A$

possiamo riscrivere le produzioni come

$S \rightarrow 012 \mid 0A12 \quad A1 \rightarrow 1A \quad A2 \rightarrow B122$

$1B \rightarrow B1 \quad 0B \rightarrow 00 \mid 00A$

Per ottenere 012 basta applicare $S \rightarrow 012$

Grammatiche tipo 1: linguaggi contestuali

Grammatica per L = $\{0^n1^n2^n, n>0\}$ V={S,A,B}, T={0,1,2}

Produzioni S→ 012 S→0A12 A1→1A A2→B122
1B→B1 0B→00 0B→ 00A

Per ottenere 001122 applichiamo (fra parentesi la produz. usata) e abbiamo (in rosso la parte destra della produzione,sottolineata la parte sinistra)

S(2)→ 0A12 (3)→ 01A2 (4)→ 01B122 (5)→ 0B1122 (6)→ 001122

Per ottenere 000111222 abbiamo (fra parentesi la produz. usata)
S (2)→ 0A12 (3)→ 01A2 (4)→ 01B122 (5)→ 0B1122(7)→
00A1122
A questo punto similmente a prima abbiamo
00A1122 (3)→ 001A122 (3)→ 0011A22 (4)→ 0011B1222 (5)→
001B11222 (5)→ 00B111222 (7)→ 000111222

Stringhe nel linguaggio

NOTA BENE. Non sempre una sequenza di derivazioni porta ad una stringa del linguaggio

essa può portare ad una forma di frase in cui non si può più applicare alcuna produzione.

Esempio. Con la grammatica precedente può accadere quanto segue (fra parentesi la produzione usata):

- S (1)⇒aSBC
  (1)⇒ aaSBCBC (2)⇒ aaaBCBCBC (4)⇒ aaabCBBCC
  (6)⇒ aaabcCBCC (3)⇒ aaabcBCCC
- A questo punto non è più applicabile alcuna produzione e la forma di frase ottenuta non darà luogo ad alcuna stringa del linguaggio.

Altri esempi

La grammatica

G=<{a, b, c}, {S,A}, P, S> con produzioni

- $S \rightarrow aSc \mid A \qquad A \rightarrow bAc \mid \varepsilon$

genera il linguaggio $\{a^n b^m c^{n+m} \mid n \geq 1, m \geq 0\}$

Idea: la prima produzione genera stringhe del tipo $a^n S c^n$ e quindi $a^n A c^n$ poi da A generiamo o la stringa vuota (ε) oppure la stringa $b^m c^m$

Grammatiche equivalenti

Def. Due grammatiche G1 e G2 sono equivalenti se generano lo stesso linguaggio - cioè $L(G1) = L(G2)$

Esempio Le grammatiche

- G1 =<{a,b},{S, A},P,S> con produzioni:

  $S \to Ab \mid b \qquad A \to aAa \mid aa$

- G2 =<{a,b},{S, A},P,S> con produzioni:

  $S \to Ab \quad A \to Aaa \mid \varepsilon$

sono equivalenti perchè generano entrambe il linguaggio $\{a^n b \mid n \geq 0 \text{ e pari}\}$.

Esercizi

Descrivere una grammatica regolare (tipo 3) che generi il linguaggio costituito da tutte e sole le stringhe binarie contenenti un numero pari di simboli 1.

Descrivere una grammatica regolare (tipo 3) che generi il linguaggio costituito da tutte e sole le stringhe binarie contenenti almeno due simboli 0 e almeno un simbolo 1.

Descrivere una grammatica libera da contesto che generi il linguaggio costituito da tutte e sole le stringhe binarie del tipo 0^n12^n (n zeri seguiti da un 1 seguito da n due).

Descrivere una grammatica libera da contesto che generi il linguaggio costituito da tutte e sole le stringhe binarie contenenti lo stesso numero di 1 e di 0 .

Descrivere una grammatica che generi il linguaggio costituito da tutte e sole le stringhe binarie contenenti un numero pari di simboli 1.

00110101   11   001100 SI

101011100   10000 NO

Grammatica (tipo 2)

V= {S}   T={0,1} produzioni

$S \rightarrow 11S \mid S11 \mid 1S1 \mid 0S \mid S0 \mid \varepsilon$

Descrivere una grammatica regolare (tipo 3) che generi il linguaggio costituito da tutte e sole le stringhe binarie contenenti un numero pari di simboli 1 (almeno 2).

Grammatica (tipo3)

V= {S,B} T={0,1} produzioni

$S \rightarrow 0S$ $S \rightarrow 1B$ $B \rightarrow 0B$ $B \rightarrow 1S$

$B \rightarrow 1C$ $C \rightarrow 0C$ $C \rightarrow \varepsilon$

- la prima produzione genera stringhe del tipo 0000S
- La seconda genera un 1 e una B (dopo i primi 0 se ci sono)
- La terza permette di generare degli 0 dopo l'1
- La quarta genera un 1 e con S si ricomincia (in altre parole le produzioni 1-4 generano sequenze di 0 e esattamente due 1 il cui ultimo carattere è S; applicando la prima produzione piu' volte otteniamo sequenze di 0 e numero pari di 1 e in cui l'ultimo carattere è S)
- Le prod. 5,6,7 permettono di chiudere generando l'ultimo 1 e gli zeri che eventualmente lo seguono

Descrivere una grammatica regolare (tipo 3) che generi il linguaggio costituito da tutte e sole le stringhe binarie contenenti un numero pari di simboli 1 ~~(almeno 2)~~. Modifichiamo la grammatica precedente

Se non ci sono 1 basta aggiungere la produzione $S \rightarrow \varepsilon$ che insieme a $S \rightarrow 0S$ permette di generare sequenze di 0

Grammatica (tipo3)

V= {S,B} T={0,1} produzioni

$S \rightarrow 0S$ $S \rightarrow 1B$ $B \rightarrow 0B$ $B \rightarrow 1S$

$B \rightarrow 1C$ $C \rightarrow 0C$ $C \rightarrow \varepsilon$

- la prima produzione genera stringhe del tipo 000S
- La seconda genera un 1 e una B (dopo i primi 0 se ci sono)
- La terza permette di generare degli 0 dopo l'1
- La quarta genera un 1 e con S si ricomincia (in altre parole le produzioni 1-4 generano sequenze di 0 e esattamente due 1 il cui ultimo carattere è S; applicando la prima produzione piu' volte otteniamo sequenze di 0 e numero pari di 1 e in cui l'ultimo carattere è S)
- Le prod. 5,6,7 permettono di chiudere generando l'ultimo 1 e gli zeri che eventualmente lo seguono

Descrivere una grammatica libera da contesto (tipo 2) che generi il linguaggio costituito da tutte e sole le stringhe binarie del tipo 0^n12^n (n zeri seguiti da un 1 seguito da n due).

012 , 0001222 SI

021 0122 NO

Grammatica 1

V= {S} T={0,1,2} produzioni

$S \rightarrow 0S2$ $S \rightarrow 1$ (in questo caso $n \geq 0$ e 1 appartiene al ling.)

Grammatica 2

V={S,A} T={0,1,2} produzioni

$S \rightarrow 0A2$ $A \rightarrow 0A2$ $A \rightarrow 1$ (in questo caso $n > 0$ e 1 non appartiene al ling.)