



# Fondamenti d'Informatica: Grammatiche

Barbara Re, Phd

# Agenda

---

- ▶ Introdurremo ...
  - ▶ Strumenti per definire un linguaggio
    - ▶ Approccio Generativo
    - ▶ Approccio Riconoscitivo

# Grammatiche

---

- ▶ Con il termine grammatica s'intende:
  - ▶ un formalismo che permette di definire un insieme di stringhe mediante l'imposizione di un particolare metodo per la loro costruzione
  - ▶ Si parte da una stringa particolare che viene via via riscritta secondo una qualche regola specificata nella grammatica
  
- ▶ Grammatica di Chomsky
  - ▶ Furono introdotte con lo scopo di rappresentare i procedimenti sintattici elementari che sono alla base della costruzione di frasi della lingua inglese
  - ▶ Si rivelano inadeguate per lo studio del linguaggio naturale
  - ▶ Giocano un ruolo fondamentale nello studio delle proprietà sintattiche dei programmi e dei linguaggi di programmazione

# Grammatica formale: definizione

---

Una grammatica formale  $G$  è una quadrupla

$$G = (V_T, V_N, P, S)$$

1.  $V_T$  è un insieme finito e non vuoto di simboli detto alfabeto terminale, i cui elementi sono detti caratteri terminali, o terminali
2.  $V_N$  è un insieme finito e non vuoto di simboli, detto alfabeto non terminale i cui elementi sono detti caratteri non terminali (o non terminali, o variabili, o categorie sintattiche)
3.  $P$  è detta insieme delle regole di produzione (o delle produzioni, o delle regole sintattiche).  $P$  è una relazione binaria di cardinalità finita su

$$(V_T \cup V_N)^* \circ V_N \circ (V_T \cup V_N)^* \times (V_T \cup V_N)^*$$

Una coppia  $\langle \alpha, \beta \rangle \in P$ , si indica generalmente con la notazione  $\alpha \rightarrow \beta$

4.  $S \in V_N$  è detto assioma ed è il simbolo non terminale di inizio, ossia la categoria sintattica più generale

# Grammatica formale: definizione

Una grammatica formale  $G$  è una quadrupla

1.  $V_T$  è l'insieme degli elementi terminali
2.  $V_N$  è l'insieme degli elementi non terminali
3.  $P$  è l'insieme delle regole sintattiche

Il linguaggio generato dalla grammatica è l'insieme delle stringhe costituite da soli terminali ai quali si può pervenire partendo dall'assioma e applicando una sequenza, arbitrariamente lunga, di passi di riscrittura

ai quali  
e i cui  
regole

Una

4.  $S \in V_N$  è detto assioma ed è il simbolo non terminale di inizio, ossia la categoria sintattica più generale



# Convenzioni

---

- ▶ Forma compatta per rappresentare le regole di produzione
- ▶ L'unione  $V_T \cup V_N$  viene indicata con  $V$
- ▶ Maiuscole dell'alfabeto latino per denotare i caratteri di  $V_N$
- ▶ Minuscole iniziali dell'alfabeto latino per denotare i caratteri di  $V_T$
- ▶ Minuscole finali per denotare stringhe di  $V_T^*$
- ▶ Minuscole dell'alfabeto greco per indicare stringhe di  $V^*$

# $\lambda$ -produzione o $\lambda$ -regola

---

Una regola del tipo  $\alpha \rightarrow \lambda$ , dove  $\alpha \in V^* \circ V_N \circ V^*$ , prende il nome di  $\lambda$ -produzione o  $\lambda$ -regola



# Derivazione diretta e derivazione

- ▶ Sia data una grammatica  $G = \langle V_T, V_N, P, S \rangle$ . La **derivazione diretta** (rispetto a  $G$ ) è una relazione su  $(V^* \circ V_N \circ V^*) \times V^*$ , rappresentata con il simbolo  $\Rightarrow$  e così definita: la coppia  $\langle \phi, \psi \rangle$  appartiene alla relazione, e scriviamo  $\phi \Rightarrow \psi$  ( $\psi$  deriva direttamente da  $\phi$  tramite  $G$ ) se esistono  $\alpha \in V^* \circ V_N \circ V^*$  e  $\beta, \gamma, \delta \in V^*$  tali che  $\phi = \gamma\alpha\delta$ ,  $\psi = \gamma\beta\delta$  e  $\alpha \rightarrow \beta \in P$
- ▶ Data una grammatica  $G$ , una **derivazione** (in  $G$ ) è una sequenza di stringhe  $\phi_1, \dots, \phi_n \in V^*$  tali che  $\forall i \in \{1, \dots, n-1\} \phi_i \Rightarrow \phi_{i+1}$

# Generazione del linguaggio

---

**Definizione 2.5** *Data una grammatica  $\mathcal{G}$ , si definisce forma di frase (in  $\mathcal{G}$ ) una qualunque stringa  $\phi \in V^*$  tale che  $S \xrightarrow[\mathcal{G}]{}^* \phi$ .*

**Definizione 2.6** *Il linguaggio generato da una grammatica  $\mathcal{G}$  è l'insieme*

$$L(\mathcal{G}) = \left\{ x \mid x \in V_T^* \wedge S \xrightarrow[\mathcal{G}]{}^* x \right\}.$$

- ▶ Il **linguaggio generato da una grammatica formale** è dunque un insieme di stringhe di caratteri **terminali**, ognuna delle quali si può ottenere a partire dall'assioma mediante l'applicazione di un **numero finito di passi di derivazione diretta**
- ▶ Equivalentemente, possiamo definire il linguaggio generato da una grammatica come l'insieme di tutte e sole le forme di frase composte da soli simboli terminali
- ▶ **Non è vero che ogni sequenza di derivazioni diretta conduce prima o poi ad una stringa del linguaggio generato dalla grammatica**



## Albero di derivazione

---

- ▶ Un **albero di derivazione** è un albero con vertici interni etichettati con non terminali, foglie etichettate con terminali e tale che se un vertice interno è etichettato con  $A$  e i suoi figli sono etichettati con  $X_1, \dots, X_k$  da sinistra a destra allora  $A \rightarrow X_1 \dots X_k$  è una produzione della grammatica
  - ▶ Le **derivazioni** possono essere rappresentate con **alberi di derivazione**
  - ▶ Un **sottoalbero** di un albero di derivazione è un vertice dell'albero con tutti i suoi discendenti, gli archi che li connettono e le etichette
-

# Equivalenza tra grammatiche

---

- ▶ Due grammatiche  $G1$  e  $G2$  si dicono **equivalenti** se  $L(G1) = L(G2)$
- ▶ In generale uno stesso linguaggio può essere generato da (infinite) grammatiche diverse.

# Tipologie di Grammatiche

---

Grammatiche di Tipo = 0 o A struttura di Frase

Grammatiche di Tipo 1 o  
Dipendenti dal Contesto

Grammatiche di Tipo 2  
o Libere dal Contesto

Grammatiche di Tipo 3  
o Regolari

# Grammatiche di tipo 0 (o a struttura di frase)

---

- ▶ Le grammatiche di tipo 0, dette anche non limitate, definiscono la classe di linguaggi più ampia possibile
- ▶ In esse le produzioni sono del tipo più generale:

$$\alpha \rightarrow \beta, \quad \alpha \in V^* \circ V_N \circ V^*, \quad \beta \in V^* \quad \text{dove } V = (V_T \cup V_N)$$

- ▶ Si noti che queste grammatiche ammettono anche derivazioni che “accorciano” le forme di frase, come ad esempio quelle che si ottengono applicando le  $\lambda$ -produzioni
- ▶ I linguaggi generabili da grammatiche di tipo 0 si dicono linguaggi di tipo 0

# Grammatiche di tipo 1 (o dipendenti dal contesto)

- ▶ Queste grammatiche, dette anche contestuali o context sensitive, ammettono qualunque regola di produzione che non riduca la lunghezza delle stringhe, cioè produzioni del tipo:

$$\alpha \rightarrow \beta, \quad \alpha \in V^* \circ V_N \circ V^*, \quad \beta \in V^+, \quad |\alpha| \leq |\beta|$$

- ▶ I linguaggi generabili da grammatiche di tipo 1 si dicono linguaggi di tipo 1, o contestuali, o context sensitive
- ▶ Il termine “linguaggio contestuale”, deriva dal fatto che, storicamente, questi linguaggi sono stati definiti da Chomsky come la classe dei linguaggi generabili da grammatiche aventi produzioni “contestuali” del tipo

$$\beta_1 A \beta_2 \rightarrow \beta_1 \gamma \beta_2, \quad A \in V_N, \beta_1, \beta_2 \in V^*, \gamma \in V^+,$$

in cui si esprime il fatto che la produzione  $A \rightarrow \gamma$  può essere applicata solo se  $A$  si trova nel contesto  $\langle \beta_1, \beta_2 \rangle$ .



# Grammatiche di tipo 2 (o libere dal contesto)

---

- ▶ Queste grammatiche, dette anche non contestuali o context free (CF), ammettono produzioni del tipo:

$$A \rightarrow \beta, \quad A \in V_N, \quad \beta \in V^+$$

cioè produzioni in cui ogni non terminale  $A$  può essere riscritto in una stringa  $\beta$  indipendentemente dal contesto in cui esso si trova

- ▶ I linguaggi generabili da grammatiche di tipo 2 vengono detti linguaggi di tipo 2 o non contestuali o context free (CF)

## Grammatiche di tipo 3 (o regolari)

---

- ▶ Queste grammatiche, dette anche lineari destre o regolari, ammettono produzioni del tipo:

$$A \rightarrow \delta, \quad A \in V_N, \quad \delta \in (V_T^\circ V_N) \cup V_T \quad (\text{regolare destra})$$

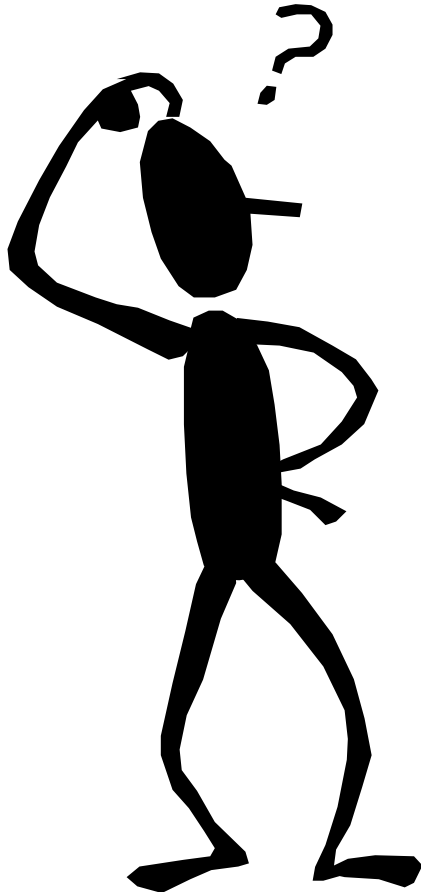
$$A \rightarrow \delta, \quad A \in V_N, \quad \delta \in (V_N^\circ V_T) \cup V_T \quad (\text{regolare sinistra})$$

- ▶ I linguaggi generabili da grammatiche di tipo 3 vengono detti linguaggi di tipo 3 o regolari

# Tipi di grammatiche

---

- Tipo 0
- $\alpha \rightarrow \beta$ ,  $\alpha \in V^* \circ V_N \circ V^*$ ,  $\beta \in V^*$  dove  $V = (V_T \cup V_N)$
- Tipo 1
- $\alpha \rightarrow \beta$ ,  $\alpha \in V^* \circ V_N \circ V^*$ ,  $\beta \in V^+$ ,  $|\alpha| \leq |\beta|$
- Tipo 2
- $A \rightarrow \beta$ ,  $A \in V_N$ ,  $\beta \in V^+$
- Tipo 3
- $A \rightarrow \delta$ ,  $A \in V_N$ ,  $\delta \in (V_T \circ V_N) \cup V_T$  (regolare destra)



Questions?