Corso di Linguaggi di Programmazione (corso A)

Docente: Giovanni Semeraro

Capitolo 3 – Linguaggi liberi da contesto e linguaggi dipendenti da contesto

Definizione di grammatica libera da contesto

- Una grammatica $G = (X, V, S, P)$ è *libera da contesto* (o *context-free* - C.F.) se, per ogni produzione,

$$v \to w$$

v è un nonterminale.

$$G \text{ è libera da contesto } \overset{def}{\Leftrightarrow} \forall\, v \to w \in P : v \in V$$

Definizione di linguaggio libero da contesto

- Un linguaggio *L* su un alfabeto *X* è *libero da contesto* se può essere generato da una grammatica libera da contesto.

  L libero da contesto $\overset{def}{\Leftrightarrow} \exists G$ libera da contesto tale che *L*(*G*) = *L*.

- Se si ha una grammatica C.F. che genera *L*, <u>non è detto che non esista</u> un'altra grammatica che generi lo stesso linguaggio.

Linguaggi liberi da contesto

- La maggior parte dei linguaggi di programmazione sono C.F.
- Il termine C.F. nasce dal fatto che la sostituzione di un *NT* non è condizionata dal contesto - ossia dai caratteri adiacenti - in cui compare.
- Un *NT* *A* in una forma di frase può sempre essere sostituito usando una produzione del tipo $A \rightarrow \beta$. La sostituzione è sempre valida.
- Viceversa, se $L = L(G)$ e G <u>non</u> è C.F., <u>non</u> possiamo concludere che L non è C.F. perché non possiamo escludere che esista una grammatica C.F. G' per cui $L=L(G')$.

Esempi di linguaggi C.F.

- Il linguaggio delle parentesi ben formate
- Il linguaggio dei numeri interi relativi
- Il linguaggio $L = \{a^n b^n \mid n > 0\}$
- Il linguaggio delle stringhe con ugual numero di 0 e di 1.
- Il linguaggio $L = \{a^n b^{2n} \mid n > 0\}$

Definizione di grammatica dipendente da contesto

- Una grammatica $G = (X, V, S, P)$ è *dipendente da contesto* (o *context-sensitive* - C.S.) se ogni produzione è in una delle seguenti forme:
  - (1) $yAz \to ywz$ con $A \in V,\ \ y, z \in (X \cup V)^*,\ \ w \in (X \cup V)^+$
    che si legge: "*A* può essere sostituita con *w* nel contesto *y*-*z* " (contesto sinistro *y* e contesto destro *z*).
  - (2) $S \to \lambda$ purché *S* non compaia nella parte destra di alcuna produzione.

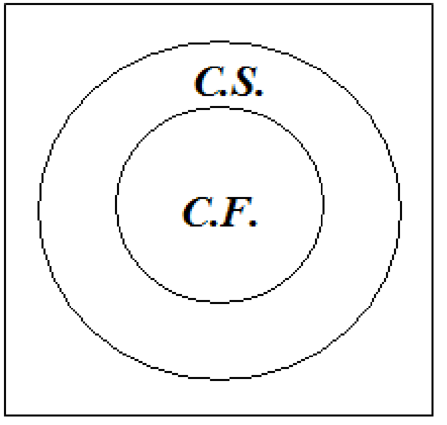
Definizione di linguaggio dipendente da contesto

- Un linguaggio *L* è *dipendente da contesto* se può essere generato da una grammatica dipendente da contesto.

$$L \text{ dipendente da contesto} \overset{def}{\Longleftrightarrow} \exists G \text{ dipendente da contesto tale che } L(G) = L.$$

Relazione tra linguaggi C.F. e C.S.

Linguaggi formali



- Tale relazione sussiste perché le regole di produzione C.S. sono una generalizzazione di quelle C.F.
- Le produzioni C.F. sono un caso particolare delle produzioni di tipo (1) delle grammatiche C.S., che si verifica quando:

$$y = z = \lambda$$

contesto destro e sinistro equivalenti alla parola vuota (c'è una eccezione).

Eccezione

- Le produzioni C.F. sono un caso particolare delle produzioni di tipo (1) delle grammatiche C.S., che si verifica quando contesto destro e sinistro sono equivalenti alla parola vuota.
  - Osservando con attenzione la definizione di grammatica C.F. si nota che, $w \in (X \cup V)^*$ mentre nella definizione di grammatica C.S. $w \in (X \cup V)^+$. Dunque le grammatiche C.F. ammettono produzioni del tipo, $A \rightarrow \lambda$ con *A* che può anche non essere il simbolo iniziale, mentre le grammatiche C.S. non ammettono tali produzioni.
  - Chiameremo tutte le produzioni del tipo *λ-produzioni* o *λ-regole*.

Esempi

- Esempi di produzioni contestuali
  - $bC \rightarrow bc$
  - $baACbA \rightarrow baAabA$
- Esempio di grammatica contestuale
  - $S \rightarrow \lambda \mid bC$

    $bC \rightarrow bc$

$S \rightarrow \lambda$ è una produzione C.S. ed *S* non compare a destra di un'altra produzione.

- Esempio di produzione non C.S. (né C.F.)
  - $CB \rightarrow BC$

non è né C.S. né C.F. È una produzione *monotona* perché del tipo $v \rightarrow w$ con $|v| \leq |w|$

Definizione di grammatica monotona

- Una grammatica G = (X, V, S, P) è *monotona* se ogni sua produzione è monotona, cioè se

$$\forall v \rightarrow w \in P : |v| \leq |w|$$

Definizione di linguaggio monotono

- Un linguaggio *L* è *monotono* se può essere generato da una grammatica monotona.

$$L \text{ monotono } \overset{def}{\Longleftrightarrow} \exists G \text{ monotona tale che } L(G) = L.$$

Esempio

- Produzioni monotone
  - $AB \rightarrow CDEF$
  - $CB \rightarrow BC$
- Una produzione monotona può essere sostituita da una sequenza di produzioni contestuali senza alterare il linguaggio generato.
  - $AB \rightarrow CDEF$ può essere sostituita dalle seguenti produzioni contestuali:
    - $AB \rightarrow AG$
    - $AG \rightarrow CG$
    - $CG \rightarrow CDEF$

Esempio

- Produzioni monotone
  - $CB \rightarrow BC$ può essere sostituita dalle seguenti produzioni contestuali:
    - $CB \rightarrow XB$
    - $XB \rightarrow XC$
    - $XC \rightarrow BC$

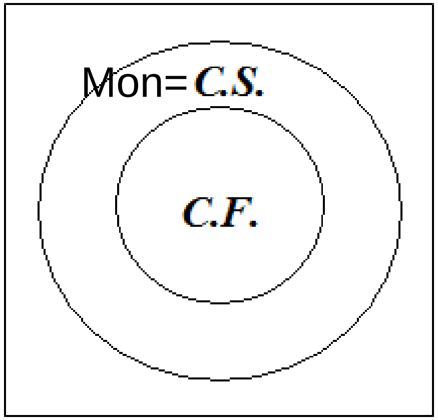
    oppure
    - $CB \rightarrow X_1B$
    - $X_1B \rightarrow X_1X_2$
    - $X_1X_2 \rightarrow X_1C$
    - $X_1C \rightarrow BC$

Proposizione

- La classe dei linguaggi contestuali coincide con la classe dei linguaggi monotoni.

C.S. = Mon

Linguaggi formali



- Tale proposizione deriva immediatamente dal teorema che segue

Teorema

- Sia *G* una grammatica monotona, cioè tale che ogni produzione di *G* è della forma $v \to w$, con $|v| \le |w|$, eccetto che ci può essere un'unica λ-produzione $S \to \lambda$ se *S* non appare alla destra di una produzione. Esiste allora una grammatica C.S. *G'* equivalente a *G*, cioè tale che *L(G)=L(G')*.
- Il teorema precedente può essere enunciato anche nella seguente forma:

Teorema (seconda formulazione)

- Un linguaggio *L* è dipendente da contesto se e solo se esiste una grammatica *G* tale che *L* = *L*(*G*) ed ogni produzione di *G* nella forma $u \rightarrow v$ ha la proprietà che: $0 < |u| \le |v|$, con una sola eccezione: se $\lambda \in L(G)$ allora $S \rightarrow \lambda$ è una produzione di *G* ed in tal caso *S* non può comparire nella parte destra di altre produzioni.

  Dimostrazione

Dimostrazione

- $\Rightarrow$) Banale.

  Se *L* è dipendente da contesto allora, per definizione, esiste *G* dipendente da contesto tale che *L* = *L*(*G*).

  $$L \text{ è C.S. } \overset{def}{\Leftrightarrow} \exists G \text{ C.S. : } L = L(G).$$

Allora ogni produzione di G è in una delle due forme:

- (1) $yAz \to ywz$ con $A \in V,\ y, z \in (X \cup V)^*,\ w \in (X \cup V)^+$
- (2) $S \to \lambda$ con *S* che non compare nella parte destra di alcuna produzione.

  Dunque, ogni produzione di *G* verifica la condizione $u \to v$, con $0 < |u| \le |v|$, se è del tipo (1), mentre se è del tipo (2) con S che non compare a destra di alcuna produzione, ricade nell'eccezione. Pertanto *G* è la grammatica cercata.

Dimostrazione

- $\Leftarrow$)
  Sia *G* una grammatica in cui ogni produzione è nella forma $u \to v$, con $0 < |u| \le |v|$. Senza ledere la generalità della dimostrazione, possiamo supporre che una generica produzione di *G* abbia il formato:
  $A_1 A_2 \ldots A_m \to B_1 B_2 \ldots B_n \qquad m \le n$ ove $A_i \in V,\ i = 1, 2, \ldots, m$
  È legittimo fare questa assunzione in quanto, se A_j fosse un terminale potremmo sostituirlo nella produzione con un nuovo nonterminale ed aggiungere la nuova produzione $A'_j \to A_j$. Denotiamo con C_1, C_2, …, C_m *m* simboli nonterminali non presenti in *G*.

Dimostrazione

$$A_1A_2\ldots A_m \rightarrow B_1B_2\ldots B_n$$

- Utilizziamo le C_k, k = 1, 2, …, m per costruire nuove regole contestuali che riscrivono la stringa $A_1A_2\ldots A_m$ con $B_1B_2\ldots B_n$.

$$\left.\begin{array}{l}
A_1A_2\ldots A_m \rightarrow C_1A_2\ldots A_m \\
C_1A_2\ldots A_m \rightarrow C_1C_2A_3\ldots A_m \\
\ldots \\
C_1C_2\ldots C_{m-1}A_m \rightarrow C_1C_2\ldots C_{m-1}C_mB_{m+1}\ldots B_n \\
\hline
C_1C_2\ldots C_{m-1}C_mB_{m+1}\ldots B_n \rightarrow C_1\ldots C_{m-1}B_mB_{m+1}\ldots B_n \\
\ldots \\
C_1B_2\ldots B_n \rightarrow B_1B_2\ldots B_n
\end{array}\right\} \begin{array}{c} 2m \\ \text{produzioni} \end{array}$$

La nuova grammatica che incorpora queste produzioni è contestuale e si può dimostrare che *L(G)=L(G')*.
Lasciamo per esercizio tale dimostrazione. *c.v.d.*

Esempio

$$\underbrace{ABC}_{m=3} \to \underbrace{DEFGH}_{n=5}$$

6 produzioni contestuali

$$ABC \to C_1BC$$

$$C_1BC \to C_1C_2C$$

$$C_1C_2C \to C_1C_2C_3GH$$

$$C_1C_2C_3GH \to C_1C_2FGH$$

$$C_1C_2FGH \to C_1EFGH$$

$$C_1EFGH \to DEFGH$$

3
+
3
produzioni
C.S.

Esercizio

- Consideriamo il linguaggio:

$$L = \{a^n b^n c^n \mid n > 0\}$$

Determiniamo una grammatica che genera tale linguaggio.

Soluzione esercizio

Riferimenti

- Semeraro, G., Elementi di Teoria dei Linguaggi Formali, ilmiolibro.it, 2017 (http://ilmiolibro.kataweb.it/libro/informatica-e-internet/317883/elementi-di-teoria-dei-linguaggi-formali/).
  - Capitolo 3