Esempio 3.3

Consideriamo il linguaggio:

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

Determiniamo una grammatica che genera tale linguaggio.

Il linguaggio somiglia ad un linguaggio già visto, $\{a^n b^n \mid n > 0\}$, generato dalla grammatica $S \to aSb \mid ab$.

Dunque le produzioni saranno del tipo:

$$S \to \overset{(1)}{aSBC} \mid \overset{(2)}{aBC}$$

il *NT B* per generare le *b*

il *NT C* per generare le *c*

Se applichiamo una volta la (1) e poi la (2), abbiamo però:

$$S \underset{(1)}{\Rightarrow} aSBC \underset{(2)}{\Rightarrow} aaBCBC$$

che non è nella forma desiderata in quanto le *b* e le *c* risulterebbero alternate.

Generalizzando, se applichiamo *n*-1 volte la (1) e poi la (2), si ha:

$$S \overset{n-1}{\underset{(1)}{\Rightarrow}} a^{n-1} S \underbrace{BCBC...BC}_{n-1} \underset{(2)}{\Rightarrow} a^n \underbrace{BCBC...BC}_{n} = a^n (BC)^n$$

Abbiamo dunque bisogno di una produzione che riporti le *B* e le *C* in posizione corretta:

$$CB \overset{(3)}{\to} BC$$

con cui:

$$S \underset{(1)}{\Rightarrow} aSBC \underset{(2)}{\Rightarrow} aaBCBC \underset{(3)}{\Rightarrow} aaBBCC = a^2 B^2 C^2$$

e

$$S \overset{n-1}{\underset{(1)}{\Rightarrow}} a^{n-1} S (BC)^{n-1} \underset{(2)}{\Rightarrow} a^n \underbrace{BCBC...BC}_{n} \underset{(3)}{\Rightarrow} a^n BBCC \underbrace{BC...BC}_{n-2} \underset{(3)}{\Rightarrow}$$

$$\underset{(3)}{\Rightarrow} a^n BBCBCC...BC \underset{(3)}{\Rightarrow} a^n BBBCCC \underbrace{BC...BC}_{n-3} \overset{*}{\underset{(3)}{\Rightarrow}} a^n B^n C^n$$

Quante volte abbiamo applicato la (3)? $\sum_{i=1}^{n-1} (n-i)$ volte.

Ora abbiamo bisogno delle produzioni che generano i terminali *b* e *c*.

Consideriamo la produzione $B \to b$. Non va bene. Perché? Perché altrimenti potremmo applicarla all'inizio di una derivazione, ottenendo stringhe del tipo:

$$a^n bCbC...bC$$

che impediscono di applicare la (3) (ed hanno bisogno di ulteriori produzioni per scambiare le *b* con le *C*) e quindi di trasformare le *B* in *b* solo dopo che le *B* hanno raggiunto la posizione corretta.

Dunque dobbiamo considerare produzioni contestuali che trasformino le B in b solo dopo che hanno raggiunto la posizione corretta:

$aB \overset{(4)}{\rightarrow} ab$ per la prima occorrenza delle B

$bB \overset{(5)}{\rightarrow} bb$ per le restanti occorrenze delle B

Analogamente per le C:

$bC \overset{(6)}{\rightarrow} bc$ per la prima occorrenza delle C

$cC \overset{(7)}{\rightarrow} cc$ per le restanti occorrenze delle C

Quindi, la grammatica G che genera $L = \{a^n b^n c^n \mid n > 0\}$ è:

$$S \rightarrow \overset{(1)}{aSBC} \mid \overset{(2)}{aBC}$$
$$CB \overset{(3)}{\rightarrow} BC$$
$$aB \overset{(4)}{\rightarrow} ab$$
$$bB \overset{(5)}{\rightarrow} bb$$
$$bC \overset{(6)}{\rightarrow} bc$$
$$cC \overset{(7)}{\rightarrow} cc$$

La grammatica è monotona, ma è facilmente determinabile una grammatica C.S. equivalente.

$$CB \overset{(3)}{\rightarrow} BC \qquad \begin{array}{l} CB \overset{(3.a)}{\rightarrow} XB \\ XB \overset{(3.b)}{\rightarrow} XC \\ XC \overset{(3.c)}{\rightarrow} BC \end{array}$$

La grammatica è comunque *non deterministica* perché non è univoca la sostituzione da operare in una forma di frase, come evidenziato dal successivo esempio.

<u>Esempio 3.5</u>

$$S \underset{(1)}{\Rightarrow} aSBC \underset{(2)}{\Rightarrow} aaBCBC \underset{(4)}{\Rightarrow} aabCBC \underset{(6)}{\Rightarrow} a^2bcBC$$

e a questo punto non possiamo più andare avanti nella derivazione perché non ci sono produzioni che possiamo applicare.

Si può dimostrare formalmente che questa grammatica genera solo stringhe di tipo $a^n\, b^n\, c^n$.

$$L = L(G)$$

Non forniamo l'intera dimostrazione, ma ci limitiamo ad osservare che:

1) $L \subseteq L(G)$ in quanto

$S \overset{n-1}{\underset{(1)}{\Rightarrow}} a^{n-1}S(BC)^{n-1}$ usando la (1) *n*-1 volte

$a^{n-1}S(BC)^{n-1} \underset{(2)}{\Rightarrow} a^n (BC)^n$ usando la (2) 1 volta

$a^n (BC)^n \overset{f(n)}{\underset{(3)}{\Rightarrow}} a^n B^n C^n$ usando la (3) *f*(*n*) volte

con $f(n) = \begin{cases} f(0) = 1 \\ f(n) = f(n-1) + n - 1 \end{cases}$

o anche $f(n) = \sum_{i=1}^{n-1} (n-i)$

$a^n B^n C^n \underset{(4)}{\Rightarrow} a^n bB^{n-1}C^n$ usando la (4) 1 volta

$a^n bB^{n-1}C^n \overset{n-1}{\underset{(5)}{\Rightarrow}} a^n b^n C^n$ usando la (5) *n*-1 volte

$a^n b^n C^n \underset{(6)}{\Rightarrow} a^n b^n cC^{n-1}$ usando la (6) 1 volta

$a^n b^n cC^{n-1} \overset{n-1}{\underset{(7)}{\Rightarrow}} a^n b^n c^n$ usando la (7) *n*-1 volte

Dunque:

$$\forall n,\ n > 0 : a^n b^n c^n \in L(G)$$

2) $L(G) \subseteq L$ per esercizio.

Dunque: $L = \{a^n b^n c^n \mid n > 0\}$ è un linguaggio C.S. (e monotono), per il teorema che stabilisce una relazione tra grammatiche C.S. e grammatiche monotone, mentre $L = \{a^n b^n \mid n > 0\}$ è C.F.