Ambiguità

di grammatiche CF

di linguaggi CF

Ambiguità nelle grammatiche e nei linguaggi

grammatiche associano una struttura ai programmi, DTD eccetera

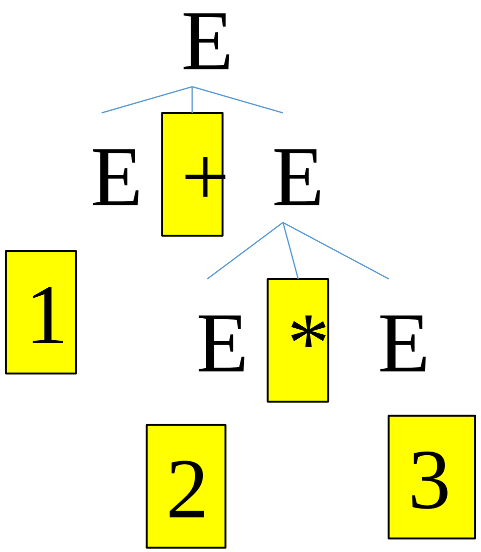
ma è una struttura univoca?

non sempre

esempio di ambiguità

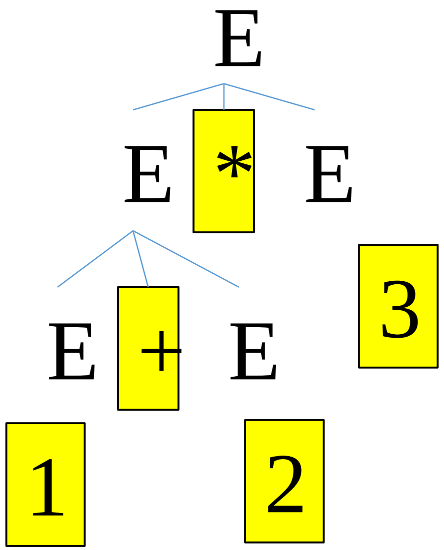
E => E+E =>E+E*E

E=>E*E => E+E*E



1 + (2*3)=7

ok



(1+2) *3 =9

sbagliato!

prodotto

1 + 2 * 3

Definizione:

Una CFG G=(V,T,R,S) è ambigua, se esiste una stringa w in T* che appartiene al linguaggio di G e per cui esistono 2 (almeno) alberi di derivazione diversi con w come prodotto.

attenzione: non derivazioni diverse ! Ma ALBERI diversi!!

Eliminare l’ambiguità di una grammatica ?

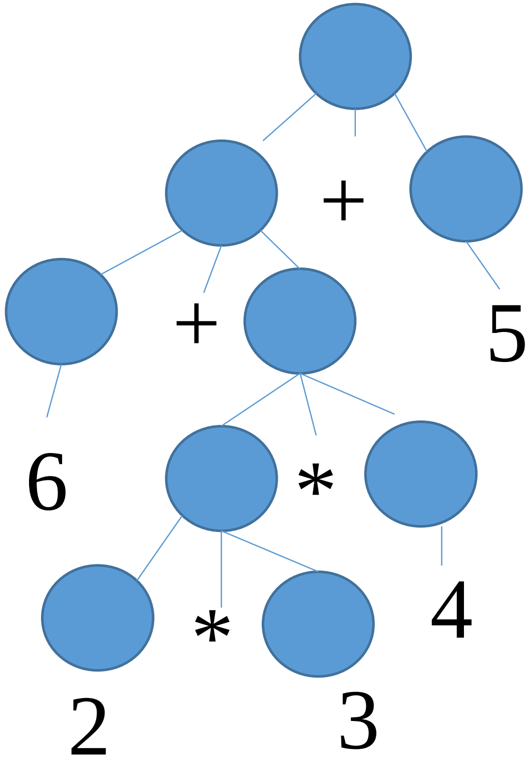
Non è sempre possibile !!
Dipende dal linguaggio che deve generare. A volte è necessario cambiare il linguaggio introducendo dei terminali che servono solo a disambiguarlo.

Nell’esempio delle espressioni notiamo che ci sono 2 cause di ambiguità:
--la precedenza degli operatori
--l’associatività degli operatori

precedenza : $* > +$ e associatività a sinistra

$6 + 2 * 3 * 4 + 5 = (6 + (2 * 3) * 4) + 5$

vogliamo una grammatica che dia la struttura giusta a tutte le espressioni



--- Si può cambiare la grammatica in modo che imponga le regole della precedenza e dell’associatività:

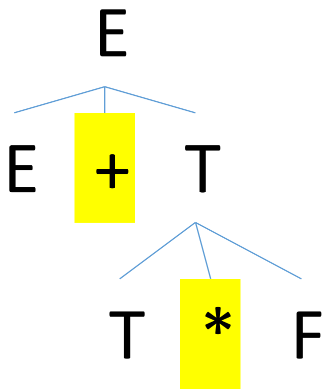
---per la precedenza basta introdurre una variabile per ogni livello di precedenza. Quelle che corrispondono a livelli di precedenza più bassi generano le altre.

I -> a | b | Ia | Ib | I0 | I1

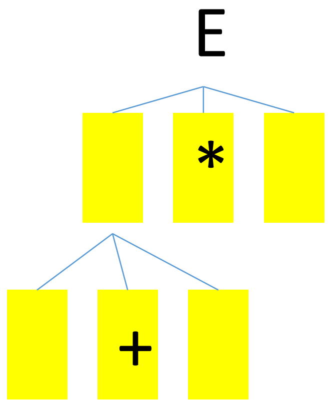
F -> I | (E)

T -> F | T * F

E -> T | E + T



SI



NO

e1 + e2 + e3 + e4

la si può generare con E -> E + T

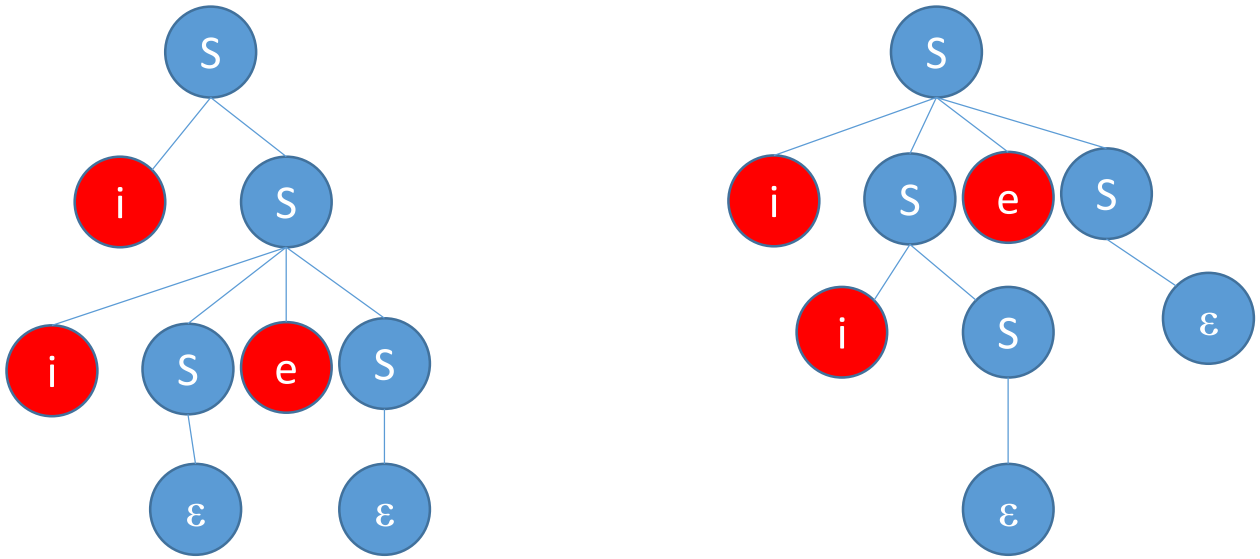
in cui T deriva e4 ed E => e1 + e2 + e3

lo stesso vale per t1 * t2 * t*3 ....

anche

S -> iS | iSeS | ε

è ambigua

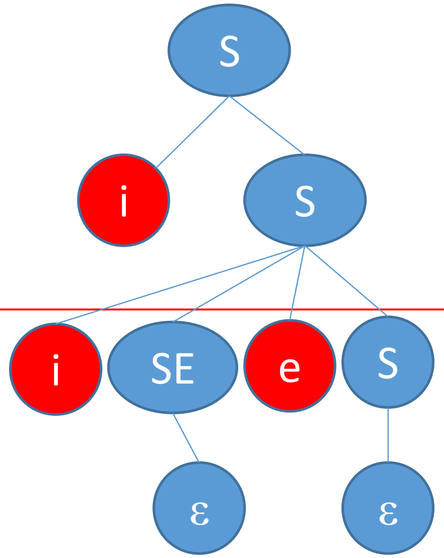


prodotto uguale i i e

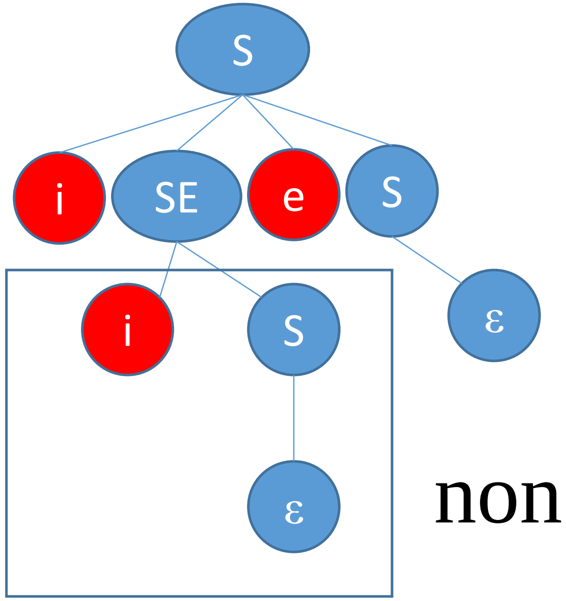
ecco una grammatica non ambigua:

S -> i S | i SE e S| ε

SE -> i SE e SE | ε



per iie



non sarebbe possibile

una grammatica è ambigua se ammette 2 alberi di derivazione diversi con uguale prodotto

2 alberi diversi => 2 derivazioni lm diverse

quindi una grammatica è ambigua se esistono 2 derivazioni lm diverse per la stessa stringa

(vale anche per rm)

Esercizio 5.4.1, 5.4.2 e 5.4.3

un linguaggio libero da contesto L è inerentemente ambiguo se ogni grammatica G tale che L = L(G) è ambigua.

inerentemente ambiguo è
$L=\{a^n b^n c^m d^m \mid n >=1, m>=1\} \cup \{a^n b^m c^m d^n \mid n >=1, m>=1\}$

S -> AB | C
A -> a A b | ab   è ambigua: considera $a^2b^2c^2d^2$
B -> c B d | cd   e non c'è niente da fare
C -> a C d | a D d
D -> b D c | bc