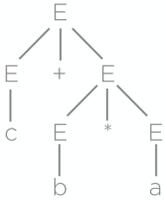


# Alberi di derivazione e ambiguità

Si possono rappresentare le derivazioni utilizzando alberi ordinali: la radice è il simbolo distinto, le foglie sono simboli terminali e tutti gli altri nodi simboli non terminali. La stringa *testimoniata* dall'albero si ottiene concatenando le foglie da sinistra a destra.



$$E \rightarrow E + E \rightarrow E + E * E \rightarrow E + E * a \rightarrow E + b * a \rightarrow c + b * a$$

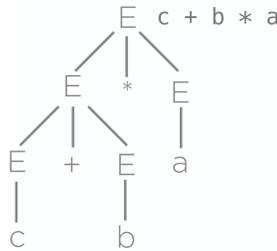
(derivazione canonica destra)

$$E \rightarrow E + E \rightarrow E + E * E \rightarrow E + b * E \rightarrow c + b * E \rightarrow c + b * a$$

(derivazione non canonica)

Una grammatica è ambigua se esiste una stringa del linguaggio generato che può essere ottenuta con due alberi di derivazione distinti con la stessa radice, ovvero con due derivazioni canoniche destre (o sinistre) distinte.

Impiegando una derivazione canonica sinistra nell'esempio sopra si ottiene la stessa stringa ma un albero diverso:

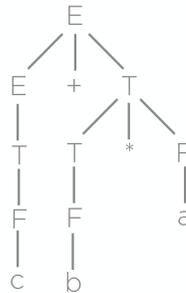


Le grammatiche non ambigue codificano la precedenza degli operatori sfruttando la profondità nell'albero:

$$E ::= E + T \mid T$$

$$T ::= T * F \mid F$$

$$F ::= a \mid \dots \mid z \mid (E)$$



Eliminando i nodi intermedi che non hanno figli foglie e rinominandoli come la radice si ottiene un **albero di sintassi astratta** uguale a quello del primo esempio, dal momento che l'ambiguità è stata risolta e l'analizzatore semantico non ha bisogno di conoscere i dettagli che hanno portato alla formazione dell'albero.