

Soluzioni Esercizi Automi

Gabriel Rovesti

June 2024

Proprietà delle Grammatiche Context Free

Esercizio 1

Considerate la seguente grammatica G che definisce un frammento di un linguaggio di programmazione:

$$\begin{aligned}\text{STMT} &\rightarrow \text{ASSIGN} \mid \text{IF-THEN} \mid \text{IF-THEN-ELSE} \\ \text{IF-THEN} &\rightarrow \text{"if condition then"} \text{ STMT} \\ \text{IF-THEN-ELSE} &\rightarrow \text{"if condition then"} \text{ STMT} \text{ "else"} \text{ STMT} \\ \text{ASSIGN} &\rightarrow \text{"a := 1"}\end{aligned}$$

Questa grammatica è ambigua perché consente più alberi di parsing per la stessa stringa di input, in particolare per le stringhe che coinvolgono costrutti if-then-else annidati. Ad esempio, consideriamo la stringa:

"if condition then if condition then a := 1 else a := 1"

Questa stringa può essere analizzata in due modi diversi a causa della presenza di più produzioni STMT che possono essere scelte per il secondo STMT nell'if-then o nell'if-then-else. Questi due parsing portano a due alberi sintattici diversi, dimostrando l'ambiguità della grammatica.

Esercizio 2

Per rendere la grammatica non ambigua, una strategia comune è modificare la grammatica per chiarire la precedenza e l'associazione delle istruzioni if-then-else. Una soluzione è introdurre una nuova non terminale per gestire l'ambiguità:

$$\begin{aligned}\text{STMT} &\rightarrow \text{ASSIGN} \mid \text{COMPLEX-IF} \\ \text{COMPLEX-IF} &\rightarrow \text{IF-THEN} \mid \text{IF-THEN-ELSE} \\ \text{IF-THEN} &\rightarrow \text{"if condition then"} \text{ COMPLEX-IF} \mid \text{"if condition then"} \text{ ASSIGN} \\ \text{IF-THEN-ELSE} &\rightarrow \text{"if condition then"} \text{ COMPLEX-IF} \text{ "else"} \text{ STMT} \mid \text{"if condition then"} \text{ ASSIGN} \text{ "else"} \text{ STMT} \\ \text{ASSIGN} &\rightarrow \text{"a := 1"}\end{aligned}$$

Questa modifica specifica che ogni istruzione if deve decidere se il suo then parte è un altro if o un'assegnazione prima di decidere cosa segue dopo else. Questo chiarisce la struttura e rimuove l'ambiguità nell'interpretazione degli if annidati.